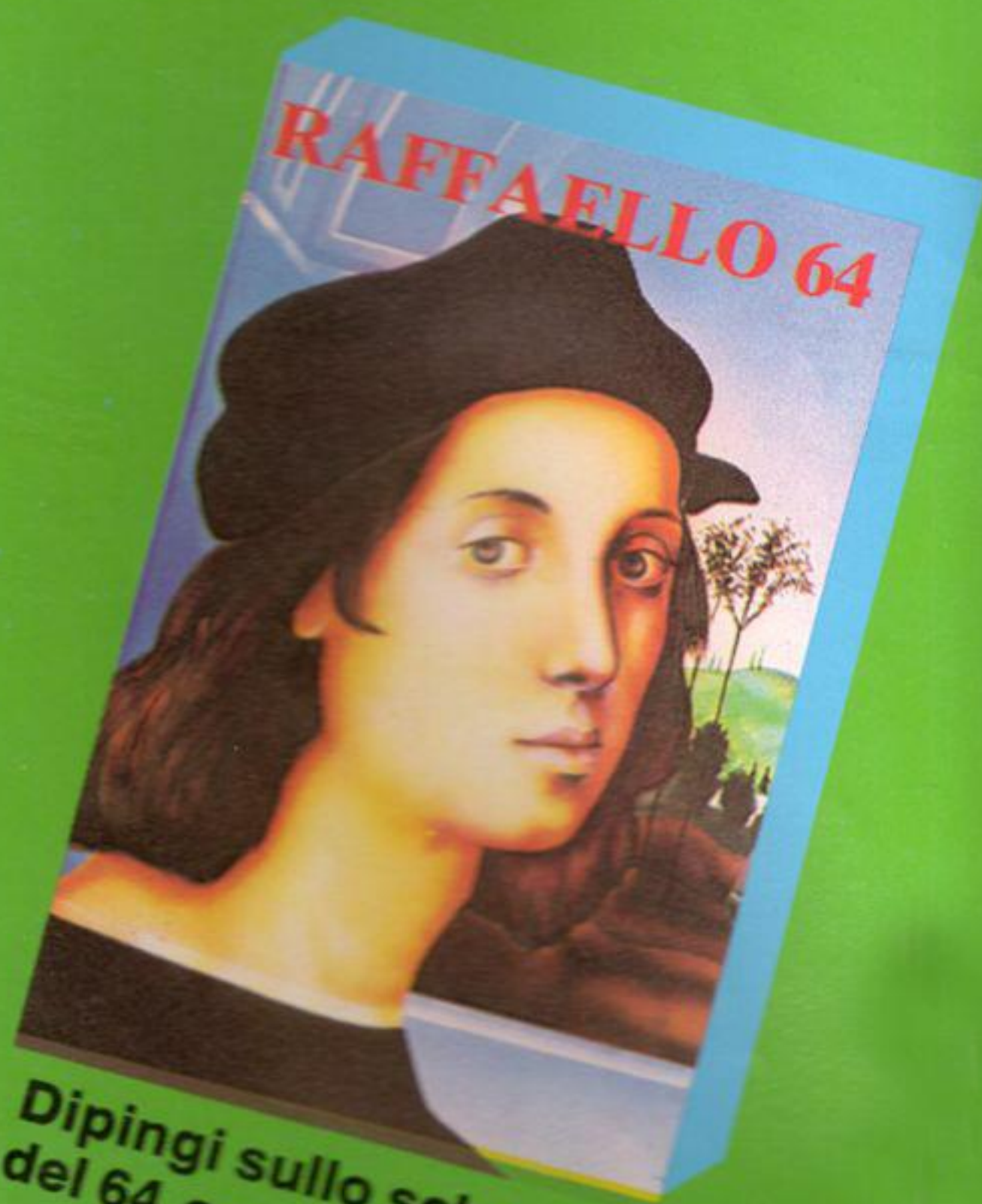
commodore
COMPUTER CLUB
21
L. 3.000
La rivista degli utenti di sistemi Commodore
Mensile - giugno 1985 - Anno IV - n° 21 - Sped. Abb. Post. Gr. III/70 - CR - Distr. MePe
Preludio musicale
Metronomo
Come fare un Commodore Club
Video, ergo chip
Kernal (ultima parte)
Nuovi supergiochi per Vic 20, C 16 e C 64
Systems

Due grandi software per il tuo 64

Impara il linguaggio macchina del 64 senza libri nè dispense

Dipingi sullo schermo del 64 senza tavoletta nè Koala, solo col tuo joystick.

In edicola

Sommario

RUBRICHE

4 LE IMMAGINI DI QUESTO FASCICOLO

8 DOMANDE/RISPOSTE

11 EDITORIALE

12 1 RIGA

18 COME CREARE UN COMPUTER CLUB

73 RECENSIONI

78 ANNUNCI ECONOMICI

PAG.	REMarks	Vic 20	C 64	C 16	Generali
	Musica				
24	Preludio musicale		•		
34	Metronomo		•		
	Giochi				
36	Il gioco del rimbalzo		•		
40	9 carte		•	•	
46	Dedalo	•			
51	Fuga!	•			
	Oltre il basic				
55	Il Kernal (ultima parte)		•		•
	L'Utile				
60	Video ergo chip		•	•	
68	Scrivi più grande sul tuo monitor		•		
	Grafica				
64	Caratteri in alta risoluzione		•		

Direttore: Alessandro de Simone
Redazione/collaboratori: Giovanni Bellù, Andrea e Alberto Boriani, Giancarlo Castagna, Eugenio Coppari, Marco De Martino, Luca Galluzzi, Giancarlo Mariani, Flavio Molinari, Enrico Scelsa, D. Matturro, M.L. Nitti, Carla Rampi, Fabio Sorgato, Danilo Toma
Segreteria di redazione: Maura Ceccaroli, Piera Perin
Ufficio Grafico: Mary Benvenuto, Arturo Ciaglia, Paolo Vertuccio
Direzione, redazione, pubblicità: V.le Famagosta, 75 - 20142 Milano - Tel. 02/8467348
Pubblicità: Milano: Mirco Croce (coordinatore), Giuseppe Porzani, Michela Prandini, Giorgio Ruffoni, Claudio Tidone, Villa Claudio - Segretaria: Lilliana Degiorgi
● Roma: Spazio Nuovo - via P. Foscari 70 - 00139 Roma - Tel. 06/8109679
Abbonamenti: Marina Vantini
Tariffe: prezzo per copia L. 3.000. Abbonamento annuo (11 fascicoli) L. 28.000. Estero: il doppio. Abbonamento cumulativo alle riviste Computer e Commodore Computer Club L. 55.000.
I versamenti vanno indirizzati a: Systems Editoriale Srl mediante assegno bancario o utilizzando il c/c postale n. 37952207
Composizioni: Systems Editoriale Srl **Fotolito:** Systems Editoriale Srl
Stampa: La Litografica S.r.l. - Busto Arsizio (VA)
Registrazione: Tribunale di Milano n. 370 del 2/1/82 - Direttore Responsabile: Michele Di Pisa
Sped. in abb. post. gr. III - Pubblicità inferiore al 70%
Distribuzione: MePe, via G. Carcano 32 - Milano

Il re Carlo I probabilmente non avrebbe gradito questa trasformazione. L'anamorfosi oltre che per finalità scientifiche e artistiche, è alla base della moderna satira caricaturale. Le immagini deformate hanno sempre fatto ridere.

GUARDA COME DONDOLO...

Il collo si allunga, la faccia si dilata, le gambe si estendono, il corpo intero si ingrossa a dismisura. Quando si entra nella sala degli specchi deformanti di un Luna Park, l'immagine riflessa della propria persona fa sempre un certo effetto. Eppure affascina, e non certo da adesso. E' dal 1600 che i più famosi pittori si sono occupati di creare questo tipo di effetti speciali.

L'esempio più eclatante si trova nella loggia del Chiostro di Trinità dei Monti, a Roma. Appena entrati, San Francesco di Paola, dipinto da Emmanuel Maignan nel 1642, sembra semplicemente in preghiera, inginnocchiato sotto a un ulivo contorto. Man mano che ci si avvicina all'affresco, però, i particolari cominciano a farsi visibili: paesaggi, navi, animali. Perfino un fiume (e qualcuno sostiene si tratti dello stretto di Messina) sul quale il santo galleggia in preghiera. Gli esempi non finiscono qui, e altri se ne possono trovare in queste pagine. La tecnica usata si chiama anamorfosi (nel caso dell'affresco di Maignan, anamorfosi prospettica).

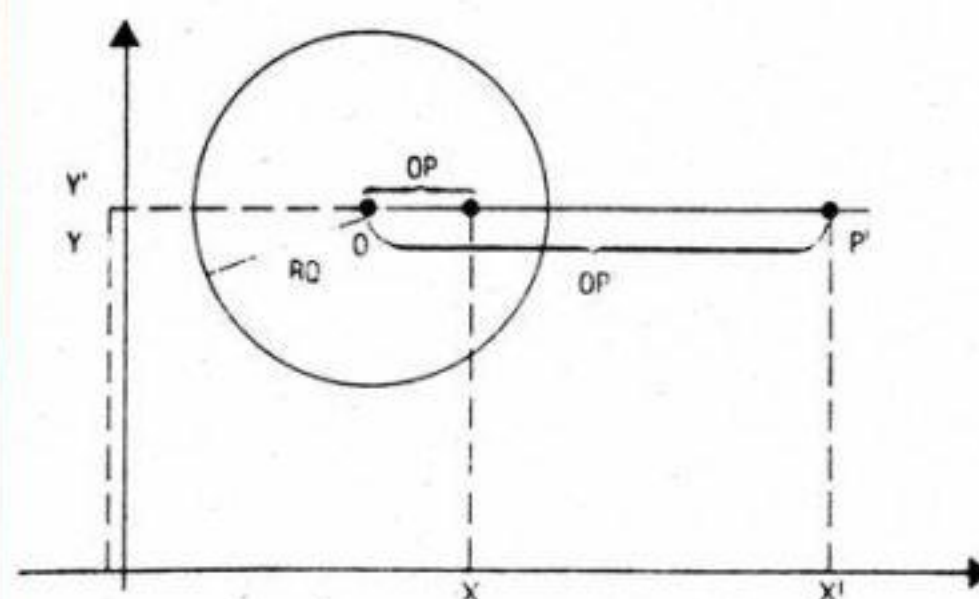
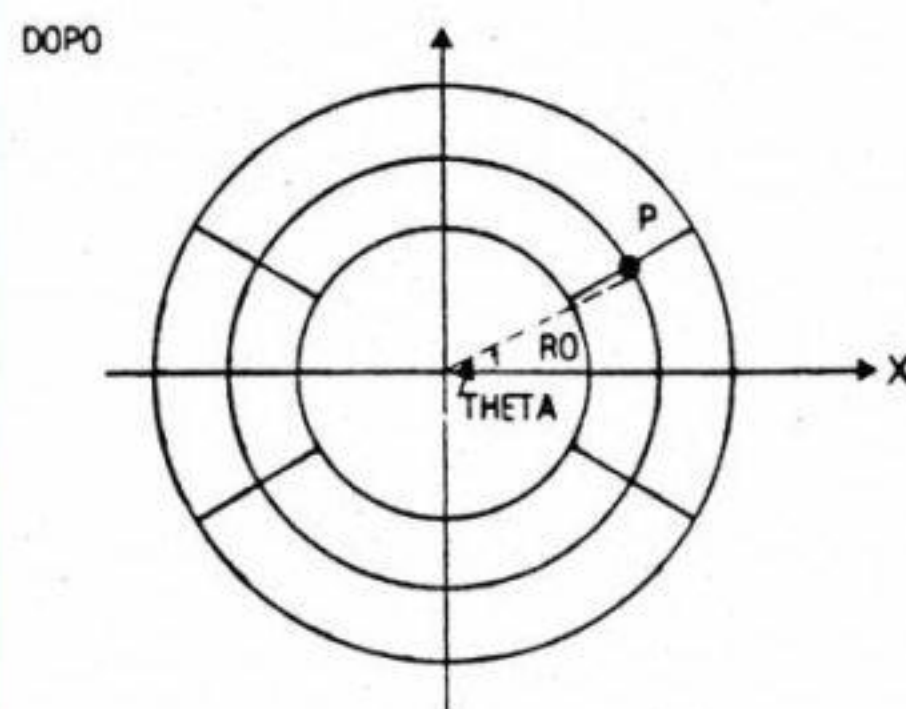
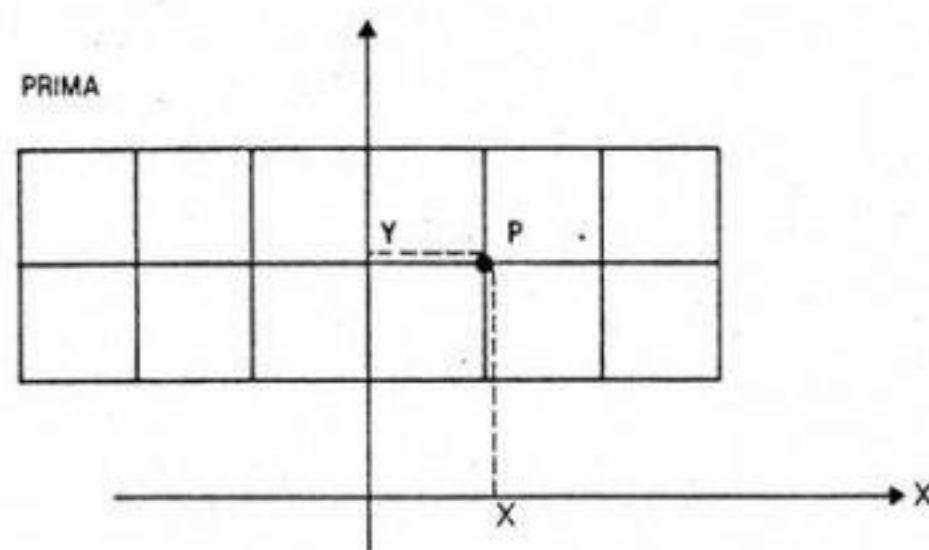
Di che cosa si tratta? Supponiamo per un attimo di lavorare su una figura disegnata su un foglio di gomma. Se il foglio viene stirato in modo da ottenere una corona circolare, con i due estremi che vengono a saldarsi, ecco che avremo creato un effetto di anamorfosi cilindrica. Se invece tratteremo il foglio di gomma come se rovesciassimo una manica di un abito (tutto ciò che sta dentro un cerchio ne esce, tutto ciò che sta fuori entra nel cerchio), ecco creato un esempio di anamorfosi conica. L'immagine è incomprensibile, ma se la guardassimo attraverso uno specchio conico si ricompone la figura originaria. Così come basterà uno specchio cilindrico appoggiato sul foglio per ristabilire l'immagine originale che avevamo deformato applicando l'anamorfosi cilindrica. Fin qui, però, si tratta di un gioco da caleidoscopio.

L'anamorfosi ha applicazioni ben più serie. Per i pittori del Seicento rappresentò un mezzo per arrivare alla definizione della tecnica della prospettiva. Attraverso una griglia quadrettata veniva dipinta l'immagine, in modo che ogni segno appartenesse a un quadretto, che diventava così il punto di riferimento per ogni trasformazione. Una tecnica rudimentale. A piccoli passi, evoluzione dopo evoluzione, si è arrivati alle applicazioni moderne delle tecniche di anamorfosi (a proposito, la parola deriva dal greco: anamorphosis significava proprio trasformazione.)

Alcuni esempi? La cartografia, dove il problema è quello di riportare su una superficie piana punti collocati su una superficie sferica. Nella fisica delle particelle e dei campi, poi, l'anamorfosi è indispensabile per portare a formule matematiche semplici equazioni che descrivono campi di forza complicati. Ma l'applicazione forse più tangibile è quella degli obiettivi gradangolari che ritroviamo sulle nostre macchine fotografiche: una visuale di 180 gradi viene "schiacciata" nello spazio di un fotogramma.

E' ovvio che l'anamorfosi è entrata a far parte anche dal patrimonio grafico dei computer. Persino dei nostri home computer, che senza arrivare alla terza dimensione riescono comunque a mostrarci esempi di anamorfosi molto convincenti. Provate ad esempio a digitare sul vostro Commodore 64 il breve programma riportato in questa pagina (attenzione: è scritto in Simon's Basic). E poi godetevi, tra un listato e l'altro di questo numero di Commodore Computer Club, le immagini create da artisti del passato e del presente giocando con l'anamorfosi.

Per una trattazione più specifica, un riferimento obbligato è al libro "Corso di grafica col computer", pubblicato da Fabbri Editori a cura di Daniele Marini. Ringraziamo la Fabbri per averci permesso la riproduzione di alcune parti del capitolo "L'anamorfosi".

```
0 REM***PROGRAMMA ANMORFOSI*************************
1 HIRES 1,0:PRINT CHR$(147)
2 DIM X(15),Y(15),U(15),V(15):A$=""
3 S1=.55:REM FATTORE SCALA CILINDRICA
4 S2=1:REM FATTORE SCALA INVERSIONE
10 GOSUB 1000:REM INIT DATI
20 GOSUB 1200:REM DISEGNA FIGURA
30 TEXT 0,0,"ANAM. C>ILINDRICA,I>NVERSIONE CONICA,
F>INE",1,1,7
35 GET A$
40 IF A$=""THEN 35
50 IF A$="F" THEN 999
60 IF A$="C" THEN 200
70 IF A$="I" THEN 500
80 GOTO 30
200 REM***ANAMORFOSI CILINDRICA*********************
210 RO=3.1415/60
220 FOR I=1 TO 15
221 X(I)=X(I)-270
222 X=(X(I)-40)*RO:Y=Y(I)-72
230 U(I)=Y*COS(X)
240 V(I)=Y*SIN(X)
250 X(I)=U(I)+160
260 Y(I)=(V(I)+180)*S1
270 NEXT I
280 GOSUB 1200
300 GOTO 30
500 REM***INVERSIONE CONICA*************************
510 RQ=15↑2:REM RAGGIO CERCHIO INVERS*************
520 FOR I=1 TO 15
525 X(I)=X(I)-160:Y(I)=Y(I)-104
530 VE=(X(I)↑2+Y(I)↑2)↑.5
532 IF VE=0 THEN VE=.0001
540 RO=RQ/VE
550 U(I)=RO*X(I)/VE
560 V(I)=RO*Y(I)/VE
570 X(I)=U(I)*S2+160
580 Y(I)=V(I)*S2+104
590 NEXT I
600 GOSUB 1200
610 GOTO 30
999 NRM:END
1000 REM***INIZIALIZZA DATI************************
1010 C=130
1020 FOR I=1 TO 5
1030 X(I)=C+15*(I-1)
1040 X(I+5)=X(I)
1050 X(I+10)=X(I)
1060 NEXT I
1070 FOR I=0 TO 2
1080 C=96+8*I
1090 FOR J=1 TO 5
1100 Y(5*I+J)=C
1110 NEXT J
1120 NEXT I
1130 RETURN
1200 REM***DISEGNA FIGURA***************************
1201 BLOCK 80,60,220,140,0
1210 FOR I=0 TO 2
1220 FOR J=1 TO 4
1230 IN=5*I+J
1240 FI=IN+1
1250 LINE X(IN),Y(IN),X(FI),Y(FI),1
1260 NEXT J
1270 NEXT I
1280 FOR I=1 TO 5
1290 LINE X(I),Y(I),X(I+5),Y(I+5),1
1300 LINE X(I+5),Y(I+5),X(I+10),Y(I+10),1
1310 NEXT I
1320 RETURN
```

IN EDICOLA

TUTTO I
IN 4

L BASIC
NASTRI

Vic 20 senza suoni

☐ **Non riesco ad ottenere il suono dal mio Vic 20. Potete aiutarmi?** *(Roberto Sangali - Villa Raverio)*

• Un segnale sonoro generato dal Vic 20 non può essere udito nei seguenti casi:
a/ locazione del volume (36878) contenente il valore nullo;
b/ contenuto delle quattro locazioni di memoria (da 36874 a 36877) inferiore a 128;
c/ difetto (ahinoi) del televisore o del computer.

Purtroppo non sappiamo se sei in grado di programmarlo correttamente. Digita pertanto, il seguente programma che ha il compito di generare un suono continuo:

```
100 POKE 36878,15
110 POKE 36874,200
```

La riga 100 pone il volume alla massima intensità sonora (argomento = 15).

La riga 110 genera un suono nel primo dei quattro generatori (quello contrassegnato da 36874).

Dopo aver dato il RUN, prova a sintonizzare il televisore fino a che non senti il suono. Questo si può interrompere premendo i tasti RUN/STOP e RESTORE, oppure col comando:

```
POKE 36878,0
```

Spostamemto del Basic

☐ **E' possibile spostare l'interprete Basic del C-64 in un'altra posizione RAM di memoria?** *(Roberto Michelotti - Genova)*

• L'interprete Basic è la zona di memoria ROM di 8-K byte (numerati da 40960 a 49151) contenente, in Linguaggio Macchina, tutte le istruzioni che consentono di interpretare correttamente i comandi e le istruzioni del linguaggio Basic.

In teoria è possibile spostare in (quasi) qualsiasi zona di memoria RAM qualunque altra zona di memoria (sia RAM che ROM). In pratica tali... migrazioni non possono portare a risultati utilizzabili.

Supponiamo, infatti, di spostare gli otto K-byte a partire dalla locazione 10960 fino a 19151. Un trasferimento puro e semplice conserva immutato il contenuto di ciascun byte con la conseguenza che gli indirizzi assoluti contenuti all'interno dell'interprete Basic sono sempre gli stessi.

In altre parole se (ad esempio) figura un'istruzione del tipo JMP (salta) alla locazione 48000, tale "salto" verrà eseguito nonstante il trasferimento. L'indirizzo 48000 dell'istruzione JMP, insomma, dovrebbe esser modificato in 18000.

Questo è solo uno dei problemi che possono presentarsi nell'effettuare spostamenti di programmi in L.M. come, appunto, deve esser considerato l'intero interprete Basic.

Un programma che, nel trasferire, tenga conto (e, di conseguenza, modifichi) gli indirizzi assoluti ed altre amenità del genere, risulta decisamente complesso e, comunque, di scarsa applicazione pratica.

Più tracce sul disco

☐ **Esistono utility per formattare 40 tracce su un disco?** *(Maurizio Belloni - Bergamo)*

• Il numero di informazioni elementari che ogni supporto magnetico può accettare dipende principalmente da due fattori:
a/ qualità del supporto magnetico stesso, vale a dire capacità di tenere sufficientemente separate (in modo da evitare dannose interferenze) zone magnetiche attigue che contengono differenti informazioni. Maggiore è tale capacità, più piccola è la zona richiesta per memorizzare un dato, più grande è la capienza del disco stesso. Un disco a doppia densità si differenzia da uno a singola densità proprio per la garanzia che può offrire nel differenziare zone attigue;
b/ qualità della meccanica del sistema di lettura/scrittura. Un drive in grado di individuare una zona magnetica molto piccola deve essere fabbricato in modo affidabile sia per ciò che riguarda la testina sia per ciò che riguarda lo spostamento radiale della testina stessa sull'apposito braccio.

In genere ogni fabbrica, tenedo conto della percentuale accettabile di errori nelle operazioni di lettura e scrittura, cerca di raggiungere un compromesso tra quantità di dati memorizzabili, velocità di operazioni, affidabilità ed economia di produzione.

Il drive 1541, per come è stato concepito e realizzato, è sufficientemente affidabile col numero noto di tracce e di settori per ciascuna traccia e con una ben assegnata velocità operativa nel senso di numero di byte trasferibili nell'unità di tempo.

Alcune Case di software hanno però realizzato, in seguito a numerose prove ed esperimenti, particolari programmi che, interagendo col sistema operativo del drive 1541, consentono di generare un maggior numero di tracce ed anche una maggior velocità di trasferimento di dati (utility di Fast Copy).

Tieni presente, però, che non è sufficiente constatare il corretto funzionamento di tali utility dopo un periodo ridotto. Un sistema si definisce "affidabile" solo dopo migliaia di prove con numerosi esemplari di dischi e drive.

Anche noi abbiamo a volte sperimentato tali utility con successo, ma non in modo sistematico tale da dare un giudizio, positivo o negativo che sia. C'è qualche lettore che ha una (lunga) esperienza con tali programmi e che ci possa fornire maggiori informazioni?

Memoria gratis

□ **C'è un modo per aumentare la memoria del Vic 20 (C-16 e C-64) tramite una semplice istruzione POKE?** *(Antonio Ranieri - Melegnano)*

• L'istruzionne POKE permette di scrivere un qualsiasi valore numerico intero compreso tra 0 e 255 in qualsiasi locazione RAM della memoria di un computer. In alcuni casi il contenuto di particolari locazioni comunica al Sistema Operativo (S.O.) il modo di comportarsi in determinate situazioni.

Ad esempio le locazioni 42 e 43 comunicano al computer il numero del byte che contiene la prima istruzione del programma Basic. Analogamente le locazioni 45 e 46 individuano l'ultima locazione del programma stesso.

Per conoscere l'indirizzo dell'ultimo byte di memoria RAM disponibile è sufficiente digitare la seguente riga:

PRINT PEEK(55)+PEEK(56)*256

Nel digitare un programma Basic si può esser tranquilli che i byte di memoria presenti dopo il valore che viene visualizzato non saranno in alcun modo modificati.

Le locazioni RAM 55 e 56 sono, appunto, chiamate "top di memoria" vale a dire il massimo di memoria disponibile.

Il contenuto di tali locazioni, gestito automaticamente dal calcolatore, possono comunque esser modificate dall'utente, dato che sono RAM, con semplici istruzioni POKE ma solo per limitare la memoria e non per aumentarla!

Se, ad esempio, vogliamo che una certa zona di memoria sia libera per allocare programmi in linguaggio macchina o per altri motivi, sarà sufficiente digitare nelle locazioni RAM 55 e 56 i due valori tali che:

PEEK(55)+PEEK(56)*256

fornisca il valore coincidente con l'ultima locazione che consentiamo sia gestita automaticamente dal computer.

In conclusione, come avrai già capito, non è possibile... creare memoria RAM con istruzioni, altrimenti nessuno avrebbe posto in commercio le apposite espansioni!

Vic 20 in produzione

□ **Il Vic 20 è ancora in produzione?** *(Vari lettori)*

• Il Vic 20 è uscito di produzione da qualche mese ed è stato sostituito egregiamente dal C-16, per ciò che riguarda la sua fascia di prezzi.

Il valore dell'usato del Vic 20 inespanso (versione base) non dovrebbe superare le centomila lire non solo perchè è obsoleto ma anche perchè richieste troppo elevate indurrebbero l'acquirente a procurarsi il nuovo C-16 decisamente più nuovo (e potente).

Colori di fondo fastidiosi

□ **Vorrei "spegnere" lo sfondo del video in modo da visualizzare solo i caratteri. Ho provato con varie combinazioni di colore ma gli occhi si stancano egualmente.** *(Andrea Tomasini - M. Urano)*

• Suppongo che per "spegnere" intendi rendere di colore nero il fondo del video. Il colore di fondo del Commodore 64 si controlla mediante il comando POKE relativo alla locazione 53281. I colori possibili sono 16 numerati da 0 a 15. Al primo (0) corrisponde il nero, al secondo il bianco. Seguono, quindi, i colori rosso, azzurro, porpora, verde, blu, giallo, arancio, marrone, rosso chiaro, grigio scuro, grigio medio, verde chiaro, blu chiaro, grigio chiaro. Volendo, ad esempio, colorare il fondo in grigio scuro è quindi necessario digitare:

POKE 53281,11

e premere il tasto Return. Se, nonostante i numerosi tentativi di selezione fondo - carattere non trovi la combinazione giusta, è probabile che il televisore che adoperi sia troppo grande o inadatto. Prova ad utilizzare un monitor a colori oppure uno in bianco - nero (monocromatico). Questi ultimi, infatti, sono più nitidi di quelli a colori.

Programmi di C.C.C. su cassetta

□ **Leggo che i programmi pubblicati su Commodore Computer Club possono essere acquistati anche su cassetta ma non specificate dove.** *(Fabio Mencucci - Marina di Carrara)*

• Solo presso la Redazione della Systems Editoriale è possibile richiedere i nastri contenenti, ciascuno, tutti i listati pubblicati su ogni fascicolo. Per richiederli è sufficiente leggere attentamente e compilare il tagliando che vedi pubblicato nell'apposito inserto pubblicitario.

Allineamento del drive

□ **Quanto sono validi i programmi che assicurano il corretto allineamento della testina del drive?** *(Sandro Natali - Sesto F.no)*

• I programmi che citi obbigano la testina stessa del drive, alterandone il sistema operativo, ad urtare contro il fine corsa tante di quelle volte fino a che non si allinea correttamente. Il sistema, forse brutale, ci è sembrato piuttosto efficace ma non lo abbiamo sperimentato frequentemente proprio per delicatezza nei confronti del povero 1541. Se qualche lettore ha riscontrato inconvenienti nell'usarlo è pregato di farcelo sapere!

Auto-run

□ **Come mai a volte è necessario battere RUN (e RETURN) per far partire un programma, mentre a volte parte automaticamente non appena termina il caricamento?** *(Luca Righetti - Dalmine)*

• I programmi su cassetta e su disco, come sicuramente saprai, dal momento in cui si caricano sul computer sono riproducibili ricorrendo al comando SAVE.

Molte ditte di software, di conseguenza, per tutelarsi almeno in parte, provvedono a porre in vendita i programmi aggiungendo alcune istruzioni particolari (non riconoscibili) per cui vengono disabilitate le funzioni di alcuni tasti (RUN/STOP, RESTORE ed altri) ed interdette le funzioni dei comandi SAVE, LIST. Per realizzare tale "trucco" si ricorre al sistema dell'auto-run, vale a dire di un particolare sistema di programmazione per cui è possibile caricare e utilizzare un programma, ma non fermarne il funzionamento ed effettuare copie dello stesso.

Per interrompere il programma, infatti, l'unico modo è quello di spegnere e riaccendere l'apparecchio ma, ahinoi!, con tale operazione la memoria del calcolatore "perde" il programma e non è più possibile effettuarne copie.

E' ovvio che il sistema di protezione descritto a nulla vale se si dispone di un copiatore di cassette, un apparecchio, cioè, che rico-

pia fedelmente su di un secondo nastro il contenuto del programma protetto.

Stampanti non Commodore

☐ **Posseggo una stampante Star con interfaccia Centronics ed un esiguo manuale di istruzioni. Nonostante molti tentativi, non riesco a farla funzionare correttamente. Come posso fare?** *(Enrico Scotti - Noverasco)*

• Non è vero che tutte le stampanti sono compatibili con tutti i computer.

Proprio per evitare dispiaceri, abbiamo più volte suggerito ai lettori di acquistare una stampante solo dopo che il negoziante avesse acconsentito ad effettuare prove non solo per verificarne la pura compatibilità hardware, ma anche (e soprattutto) la compatibilità software: alcuni programmi, come word processor o data base, non funzionano correttamente se è inserita sul retro del calcolatore un'interfaccia, pur se originale Commodore.

Una risposta seria alla tua domanda richiederebbe, come puoi intuire, uno studio "dal vero" dell'apparecchio. Mi dispiace, ma non sappiamo proprio come venirti in aiuto dato che non abbiamo una esperienza diretta con la stampante.

C-16 e tasto HELP

☐ **Che funzione ha il tasto HELP sul C-16 e Plus-4?** *(Nicola Zanella - Trento)*

• Il tasto di cui parli è di notevole utilità soprattutto per i principianti che, in alcuni casi, non riescono a individuare errori di digitazione o di programmazione.

Supponiamo che, nel digitare un programma, si commettano alcuni errori di battitura come nell'esempio che segue:

100 PRINT:PRONT:PRINT

in cui la seconda istruzione è stata digitata come PRONT invece che PRINT.

Facendo partire il programma (RUN) l'esecuzione si blocca segnalando, appunto, un ?SYNTAX ERROR IN 100.

Il computer, cioè, si limita a comunicare soltanto il numero di linea Basic in cui è stato riscontrato l'errore. Nel caso appena accennato è semplice individuarlo (digitando LIST 100 ed esaminando le istruzioni) ma, molto spesso, l'errore commesso è difficile da "scoprire".

Pertanto se, dopo la segnalazione di errore (?SYNTAX ERROR IN 100), si preme il tasto HELP, verrà visualizzata sullo schermo la riga 100 come se fosse stato impartito il comando LIST 100. L'istruzione contenente l'errore segnalato, però, apparirà "lampeggiante" facilitando, in tal modo, la correzione.

Aumento di velocità nei videogiochi

☐ **Vorrei sapere come fare per muovere molti oggetti contemporaneamente sullo schermo senza rallentare di molto un videogioco per Vic 20 e CBM 64.** *(Massimo Fortin - Milano)*

• Nei videogiochi in cui è necessario spostare immagini (dal semplice carattere a gruppi di caratteri formanti un'immagine più grande), la tecnica seguita deve essere la seguente, supponendo, per semplificare, che si voglia effettuare uno spostamento a sinistra:

a/ memorizzare, in una zona di memoria "libera", la parte video che sarà tra breve "occupata" dall'oggetto in movimento verso sinistra;

b/ spostare l'oggetto a sinistra (ciò provoca, ovviamente la cancellazione di ciò che è presente in tale zona);

c/ fare apparire, al posto occupato dall'oggetto prima dello spostamento, la zona video precedentemente memorizzata in modo analogo al punto "a".

Senza il punto "c", infatti, l'oggetto lascerebbe una 'IA" (oppure l'immagine di se stesso) nel percorrere lo schermo. Se tieni conto che quanto detto deve essere realizzato anche per gli altri spostamenti e che, contemporaneamente, è necessario controllare le mosse del giocatore (se preme oppure no qualche tasto del joystick) e generare, magari, effetti sonori, risulta chiaro che la velocità del gioco diminuisce sensibilmente.

Per evitare un eccessivo rallentamento è indispensabile ricorrere al Linguaggio Macchina (L.M.) che, evitando la lentezza tipica del linguaggio Basic, consente di ottenere gli stessi effetti a velocità migliaia di volte superiori. Ecco perchè i videogiochi in commercio di un certo "livello" sono tutti in L.M.

Purtroppo non tutti gli utilizzatori di un personal computer sono in grado di programmare in L.M.

Per venire incontro ad un pubblico più vasto sono stati creati gli sprite che sono immagini programmate spostabili, con comandi Basic alla portata di tutti, in un punto qualsiasi dello schermo senza cancellare ciò che eventualmente già è presente sul video e senza lasciare scie.

La possibilità di creare tali immagini programmabili (veri e propri microschermi all'interno del video) è però limitata al Commodore 64; il Vic 20, il C-16 ed il Plus 4 non possiedono tale caratteristica.

Nonostante le facilitazioni derivanti dall'uso degli sprite, è possibile che anche programmando col Commodore 64 la velocità non sia soddisfacente. In tali casi, come abbiamo detto, l'ostacolo si supera programmando in L.M. oppure ricorrendo programmi compilatori (Petspeed, Austrospeed) che trasformano parzialmente un programma Basic in L.M.

I compilatori in commercio, però, funzionano solo col C-64 insieme al Floppy Disk Drive 1541. Programmi Basic scritti col Vic 20 ed il C-16, almeno per ora, non possono esser compilati.

Registratore che non legge

☐ **Col mio registratore, durante il caricamento di alcuni programmi, è necessario tener premuto il coperchio che alloggia la cassetta, altrimenti i programmi non vengono letti. E' normale?** *(Fabio Morara - Bologna)*

• No: è probabile che la testina del tuo registratore non sia allineata come dovrebbe e, di conseguenza, la pressione sullo sportellino, spostando di alcuni decimi di millimetro il nastro o la testina di lettura, ripristini (fortuitamente) le condizioni ottimali.

In casi come questo (difficoltà di caricamento) è comunque necessario fare sempre le seguenti prove:

a/ allontanare il registratore dal televisore: i campi magnetici da questo provocati possono dare molto "fastidio" al caricamento di alcuni programmi;

b/ assicurarsi che il difetto risieda nel proprio registratore: il nastro potrebbe essere stato registrato con un registratore disallineato;

c/ ruotare la vite di Azimut della testina di non più di un quarto di giro a destra oppure a sinistra ed effettuare tentativi fino a caricamento corretto.

EDITORIALE · EDITOR

ORIALE · EDITORIALE

Il grande balzo in avanti

Nella testata di questa rivista figura, ben evidenziata, la parola **Club.** Con tale termine si intende comunemente un gruppo di persone che, animate da comuni interessi, si incontrano periodicamente per scambiarsi informazioni, idee, notizie.

Finora, modestia a parte, siamo riusciti sia nell'intento di colmare le lacune tipiche dei principianti, sia nell'affrontare argomenti di difficile approccio, sia nel proporre listati e procedure che consentissero di utilizzare realmente il calcolatore.

Quando la rivista è nata (prima nel suo genere), era nostra intenzione rappresentare un punto di riferimento per gli utilizzatori dell'"oggetto" calcolatore che lo avessero appena acquistato oppure lo possedessero da molto tempo.

La nascita delle due testate "sorelle" (**Commodore,** su carta e **Commodore Club,** su nastro), ci permetteva, in seguito, di accontentare non solo coloro che volevano affrontare argomenti fuori dell'usuale, ma anche utilizzatori pigri, affamati tuttavia, di programmi sofisticati da utilizzare subito, senza digitarli.

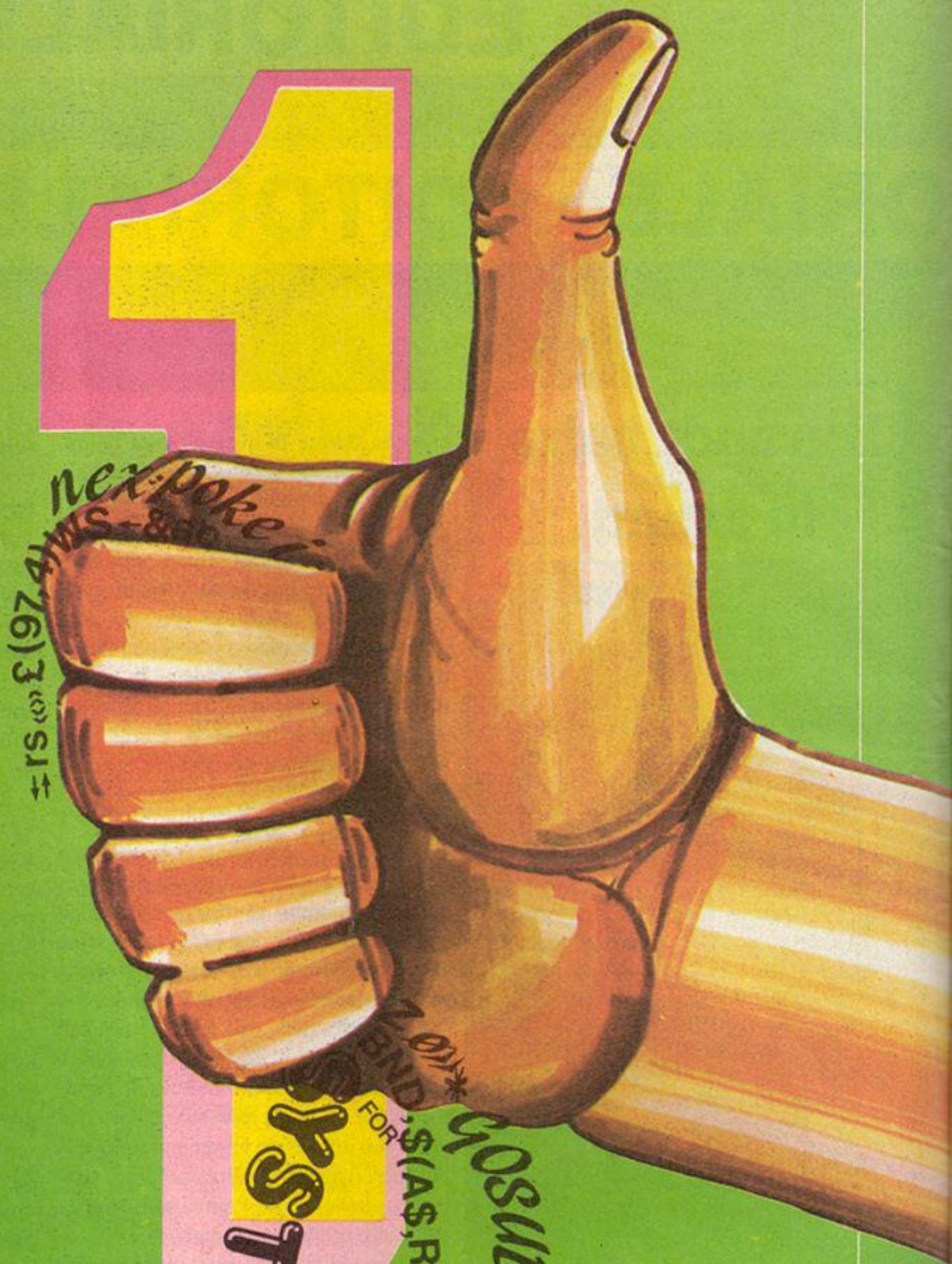
La periodicità (mensile) indispensabile per selezionare, esaminare, commissionare programmi, argomenti, problemi, è risultata, per alcuni utilizzatori... entusiasti, tanto modesta da acquistare anche altre testate e libri, ricevendo molto spesso, in cambio, delusioni e, talvolta, autentiche fregature nel trovare grossolani errori, inspiegabili doppioni, identici videogiochi presentati, però, con nomi diversi su cassette di "marca" diversa.

Avvertiamo, quindi, tra gli utenti, l'esigenza di far chiarezza nel mercato "facile" di copiatori ed informatici dell'ultima ora allo scopo non tanto di attirare dalla nostra parte i lettori potenziali (business is business), ma soprattutto per conservare interessante il campo del personal computer in generale e di Commodore in particolare.

Se, infatti, un utente deluso abbandona il computer in cantina, a rimetterci sono anche coloro che hanno sempre creduto nell'hobby (o professione?) del personal e che, nel riscontrare l'elevato numero di entusiasti, provano il desiderio di andare avanti, di proporre nuove possibilità di impiego, di essere, insomma, giustamente protagonisti in una realtà sociale che troppo spesso abitua a stare dalla parte del consumo più che della produzione.

E' con tale spirito che, in altra parte del presente fascicolo, troverai il nuovo modo di tener vivo l'interesse per il tuo personal:

"**Come creare un Computer Club**", infatti, oltre ad essere un articolo di interesse generale, vuole comunicare che i tempi son maturi per un nuovo, meraviglioso, grande balzo in avanti.

Alessandro De Simone

GOSUB
RIGA

#1

Reset/load. Quando digitiamo LOAD e premiamo il tasto RETURN, sul nostro monitor compare PRESS PLAY ON TAPE e il calcolatore è posto in fase di caricamento da registratore. Come mai si verifica ciò? Quando battiamo LOAD, il calcolatore ricerca nelle locazioni di memoria 816 e 817 il valore della cella di memoria da cui inizia la routine di caricamento del Commodore 64. Se noi modifichiamo tali valori, potremo ottenere degli effetti diversi rispetto a quelli che si verificano normalmente. Inserendo rispettivamente 226 e 252 nelle locazioni 816 e 817 il computer farà riferimanto, nel caso che si digiti LOAD, alla routine di System Reset (64738). Otterremo quindi un risultato analogo a quello che si verificherebbe digitando SYS 64738. Attraverso questo semplice e divertente esempio, dovrebbe essere più semplice comprendere la logica del comportamento del computer quando gli impartiamo un comando BASIC.

```
1 POKE 816,226:POKE 817
 ,252
```

#2

Pagina video. La locazione 648 indica la pagina video da cui inizia la memoria schermo.

In condizioni normali di funzionamento troveremo in questa cella di memoria il valore 4, corrispondente alla quarta pagina. Una pagina è costituita da 256 byte, da ciò ne dedurremo che la numero 4 avrà inizio a 1024 e termine a 1279.

Se nella locazione di memoria precedentemente citata inseriamo il valore 5, il nostro commodore 64 farà riferimento a 1280 come prima locazione video. Dopo aver dato il RUN a questa breve routine, sarà quindi impossibile spostarsi con i tasti cursore nella zona compresa tra 1024 e 1279.

```
1 POKE 648,5
```

#3

Repeat. Per poter visualizzare innumerevoli volte un carattere sul proprio video, senza essere obbligati a premere ripetutamente un determinato tasto, dovete immettere il valore 128 nella locazione 650.

```
1 POKE 650,128
```

#4

Buffer. La locazione di memoria 649 consente di dimensionare il buffer di tastiera. Normalmente in questa locazione è contenuto il valore 10. Se in essa viene inserito il numero 0, non sarà più possibile immettere alcun dato attraverso la tastiera. Dopo aver dato il RUN alla routine, non potrete più visualizzare alcun carattere sul video.

```
1 POKE 649,0
```

#5

Colonna. Tramite la locazione di memoria 211 è possibile conoscere la colonna in cui è posizionato il cursore durante una fase di esecuzione di un programma. La routine che vi proponiamo fornirà un esempio di quanto asserito.

```
1 INPUT A$:PRINTA$;:PRI
  NTPEEK(211)
```

#6

Integrali. Questa routine vi consentirà di calcolare l'integrale definito di una funzione da voi precedentemente scelta.

In fase di INPUT vi verranno richiesti gli estremi di integrazione (A e B) e il numero degli intervalli su cui effettuare il calcolo (N).

Ricordatevi di fornire tutti e 3 i dati richiesti.

```
1 DEF FNF(X)=X↑2/5+2:IN
  PUT A,B,N:D=(B-A)/N:F
  OR I=0 TO N-1:S=D*FNF
  (A+I*D+D/2)+S:NEXT:PR
  INTS
```

#7

Gradi/radianti. Questo programma effetterà la conversione da gradi in radianti.

```
1 INPUT C$,A:B=-(180*A)
  *(C$="G")-(A/180)*(C$
  ="R"):PRINTC$"="B:GOT
  O 1
```

#8

Binario/decimale. La routine effettua la conversione dal sistema binario a quello decimale.

```
1 INPUT B$:A=LEN(B$):FO
  R I=1 TO A:N=N-2↑(A-I
  )*(MID$(B$,I,1)="1"):
  NEXT:PRINTN: RUN1
```

#9

Decimale/binario. Questo programma sortisce un effetto opposto rispetto alla routine precedente.

```
1 INPUT N:FOR I=1 TO LO
  G(N)/LOG(2):Q=INT(N/2
  ):R=N-2*Q:B$=STR$(R)+
  B$:N=Q:NEXT:PRINT1;B$
  : RUN
```

#10

Micro Game. Anche una sola riga può essere sufficiente per creare un piccolo gioco.

Utilizzando i tasti funzione F1 e F3 vi muoverete con il carattere A sul vostro video e dovrete cercare di intercettare il maggior numero possibile di I.

```
1 W=1027+X:P=P-(PEEK(W)
  =9):POKE W,1:POKE 911
  +W,9:A=PEEK(203):X=X+
  (A=4)-(A=5):PRINTP:GO
  TO 1
```

CARATTERI E CODICI DEL C 64

Simbolo	Tasti utilizzati	Denominaz.	Simbolo	Tasti utilizzati	Denominaz.	Simbolo	Tasti utilizzati	Denominaz.	Simbolo	Tasti utilizzati	Denominaz.
	Shift Clr/Home	Clear		Ctrl 1	Nero		Commodore 1	Aranc		F1	F1
	Clr/Home	Home		Ctrl 2	Bianco		Commodore 2	Marr		Shift F1	F2
	Shift Crsr (S)	Up		Ctrl 3	Rosso		Commodore 3	Rosa		F3	F3
	Crsr (S)	Down		Ctrl 4	Azzur		Commodore 4	Grigio 1		Shift F3	F4
	Shift Crsr (D)	Left		Ctrl 5	Viola		Commodore 5	Grigio 2		F5	F5
	Crsr (D)	Right		Ctrl 6	Verde		Commodore 6	Verde 2		Shift F5	F6
	Ctrl 9	Rvs		Ctrl 7	Bleu		Commodore 7	Celeste		F7	F7
	Ctrl 0	Rvoff		Ctrl 8	Giallo		Commodore 8	Grigio 3		Shift F7	F8

#11

Primalità. Questa piccola routine vi consentirà di appurare la primalità di un determinato numero da voi prescelto.

```
0 INPUT N:FOR I=2 TO SQ
  R(N):A=A+(INT(N/I)=N/
  I):NEXT:A$(I)="NON":P
  RINTA$((A=0)+I)" E'":
  GOTO
```

#12

Sinusoide. Una sprite creata casualmente si muove con andamento sinusoidale sul vostro video.

```
1 V=53248:POKE V+21,1:P
  OKE 2040,1:FOR X=9 TO
   255:POKE V,X:POKE V+
  1,ABS(SIN(X/30))*190+
  4:NEXT:GOTO 1
```

#13

Secondo grado. Mediante questa routine potrete calcolare le radici reali di un'equazione di secondo grado. Dovete fornire in INPUT i coefficenti A, B e C. Nel caso che l'equazione ammetta radici immaginarie, comparirà il messaggio di errore ILLEGAL QUANTITY ERROR.

```
1 INPUT A,B,C:S=SQR(B↑2
  -4*A*C):D=2*A:PRINT"X
  1="(-B-S)/D:PRINT"X2=
  "(-B+S)/D:GOTO 1
```

#14

Indovina numero. Questa routine genera un numero casuale compreso tra 1 e 9000. Avete a disposizione 20 tentativi per individuarlo e ogni volta vi verrà segnalato se il dato fornito è troppo alto o troppo basso.

```
1 A=INT(RND(1)*9000):FO
  R I=1 TO 20:INPUT P:C
  = SGN(A-P):IF C THEN
  PRINTI,CHR$(65-(C=1))
  :NEXT
```

#15

Divisione. Questa routine vi consente di calcolare un predeterminato numero di decimali di una divisione. Fornite in fase di INPUT i due operandi e il numero di elementi che desiderate ottenere dal calcolo impostato.

```
1 INPUT N,M,D:K=INT(N/M
  ):PRINTK",";:FOR C=1
  TO D:N=(N-M*K)*10:K=I
  NT(N/M):PRINTK;:NEXT:
  PRINT: RUN
```

#16

Grump. Digitando e dando il RUN a questa routine otterrete un simpatico effetto grafico.

```
1 PRINT"[CLEAR]":A$="CA
  VALLO":FOR I=7 TO 1 S
  TEP -1:PRINTSPC(121)R
  IGHT$(A$,I);:NEXT
```

#17

Riga colonna. Un carattere comparirà casualmente sul video, cercate di indovinare la riga e la colonna a cui esso appartiene.

```
1 PRINT"[CLEAR]":A=INT(
  920*RND(1))+1:POKE 11
  03+A,0: INPUT X,Y:IF A
  +120=X+40*Y THEN PRIN
  T"BRAVO"
```

#18

Invertitore. Inserite in fase di INPUT un frase di vostro gradimento e otterrete in OUTPUT il suo esatto inverso.

```
1 INPUT N$:IF N$>"" THE
  N A=LEN(N$):FOR T=1 T
  O A:PRINTMID$(N$,A+1-
  T,1);:NEXT:PRINT:GOTO
  1
```

#19

Quadrato quiz. Il computer vi fornirà un numero e voi dovrete calcolarne il quadrato. Il gioco continuerà con difficoltà progressivamente crescente.

```
1 A=INT(RND(1)*5)+N:PRI
  NTA:INPUT Q:C=A*A=Q:P
  RINTA*A:N=N-C*3:P=P-C
  *2-1:PRINT"PUNTI="P:G
  OTO 1
```

#20

Divisore/mult. Fornendo 2 operandi verranno effettuate le operazioni di divisione e moltiplicazione tra essi.

```
0 INPUT A,B:FOR I=1 TO
  A:M=M-((INT(A/I)=A/I)
   AND (INT(B/I)=B/I))*
  (I-M):NEXT:PRINTM,A*B
  /M
```

#21

Giostra. Un variopinto alternarsi di caratteri vivacizzerà il vostro video.

```
1 FOR X=0 TO 39:POKE 14
  64+X-40*ABS((X AND 7)
  +1+8*((X AND 7)>3)),8
  1+21*((X+A) AND 1):NE
  XT:A=A+1:GOTO 1
```

#22

Stelle di agosto. Anche sul vostro computer è possibile creare una stupenda stellata notturna.

```
1 PRINT"[CLEAR]":FOR J=
  1 TO 99:Z=INT(1000*RN
  D(1)):POKE 55296+Z,IN
  T(RND(1)*15):POKE 102
  4+Z,42:NEXTJ:GOTO 1
```

Le proposte del n. 1

PER STAMPARE CON POCHE LIRE

Le stampanti **MT/85, a 80 colonne, e MT/86, a 136 colonne,** rappresentano una nuova frontiera nel settore delle stampanti a basso costo.
Basso costo, ma non bassa qualità e basse prestazioni, infatti ecco le credenziali di questi due nuovi prodotti.
Velocità a 180 cps. bidirezionale ottimizzata, NLQ a 45 cps., grafiche, possibilità di 8 fonti alternative di caratteri e naturalmente la completa compatibilità con il PC IBM.
Il prezzo: il più competitivo del mercato in questa fascia di prestazioni.
Naturalmente anche le MT/85/86 oltre ai trattori hanno anche trascinamento a frizione e consentono pertanto il trattamento del foglio singolo.

20094 Corsico (MI) - Via Cadamosto, 3
Tel. (02) 4502850/855/860/865/870 - Telex 311371 Tally I
00137 Roma - Via I. Del Lungo, 42 - Tel. (06) 8278458
10099 San Mauro (TO) - Via Casale, 308 - Tel. (011) 8225171
40050 Monteveglio (BO) - Via Einstein, 5 - Tel. (051) 832508

adverteam

PERCHE' NON DEVI PERDERE IL PROSSIMO FASCICOLO

Ecco cosa ti prepara il tuo Commodore Computer Club tutto estivo:

1/ Un simulatore di computer da usare sulla spiaggia con le conchiglie o i sassolini.

2/ Un quiz super-commodoroso per giocare sotto l'ombrellone.

3/ La scatola magica che impara dalle tue risposte.

4/ Tanti giochi ed articoli didattici.

Come Creare un Computer Club

di **Alessandro De Simone**

Scambi di software e riviste, uso in comune di computer anche costosi, un ambiente rilassato e competente. I vantaggi della partecipazione.

In ogni parte del mondo quando un hobby, uno sport o un gruppo musicale attanagliano le menti di un numero apprezzabile di sostenitori, ecco che si avverte l'esigenza di fondare un club allo scopo di incontrarsi per scambiarsi idee, esaminare progetti, proporre iniziative.

L'Italia, come è noto, non è da meno se si considera che, quando si è in quattro a pensarla allo stesso modo, c'è sempre qualcuno che suggerisce di fondare un partito.

Lungi, comunque, da velleità politiche gli entusiasti di un passatempo "tecnico" aderiscono di buon grado alle proposte di un club ed i motivi sono più che evidenti, nel nostro come in tanti altri casi analoghi:

- facilità di scambiare software direttamente su supporto magnetico allo scopo di arricchire la propria biblioteca (io do una cosa a te, tu dai una cosa a me...);
- lettura di riviste e libri che, acquistati in copia singola e messi a disposizione dei soci, consentano risparmi considerevoli e limitazioni di perdite in caso di "fregature" (doppioni, errori, pubblicazioni di S/W superato);
- possibilità di uso di apparecchi costosi acquistati in comune oppure messi, pur se per poco tempo, a disposizione dei soci da sponsor che, intelligentemente, non possono che riconoscere valida una promozione di tale tipo;
- possibilità di ottenere sconti per acquisti in quantità di nastri, dischi, libri e, perchè no?, drive, stampanti & affini;
- un luogo in cui incontrarsi senza la "rottura" di interferenze di familiari insofferenti (Ma stai sempre alla tastiera? Vieni, che si fredda la cena ...);
- un luogo "sicuro" lontano da pericolose, stupide tentazioni;
- un ambiente in cui evolversi seguendo i più bravi e incoraggiando i neofiti;
- un modo di proporre il proprio nominativo, per realizzare programmi personalizzati, a manager e professionisti desiderosi di Data Base o semplicemente di imparare ad usare il proprio calcolatore. Non avete idea di quanti commercialisti, medici, ingegneri siano disposti, dietro compenso affatto trascurabile, a seguire "lezioni private" di informatica;
- un luogo in cui proporre la vendita e la permuta delle proprie apparecchiature.

Il luogo

Il primo problema da affrontare è indubbiamente il luogo da destinare alle indispensabili riunioni.

Rinunciate immediatamente (e insospettitevi altrettanto in fretta se ve lo propongono) all'idea di aprire un club per corrispondenza, magari servendosi di un'anonima casella postale. Tale sistema, infatti, non solo è un controsenso che scoraggia l'aspirante socio (velocità

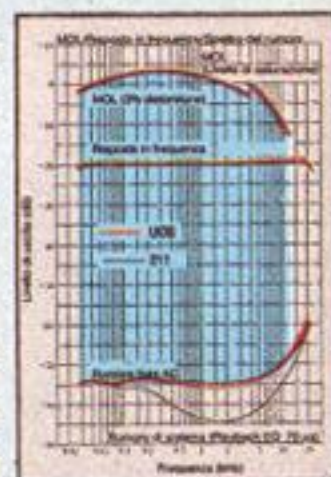
Avevano cominciato come club di appassionati: questo è il marchio che si erano dati. Adesso la Electronic Arts è una delle software house più importanti nella programmazione per personal computer. La loro filosofia: fornire un ambiente ideale ai progettisti di software indipendenti. Foto di gruppo dei fondatori. Da destra in alto: Bill Budge (Pinball Construction Set la sua creatura più famosa); Anne Westfall e John Freeman (Archon e Murder on the Zinderneuf); Dan Bunten (con la Softscape creò la prima versione di M.U.L.E.). A sinistra, Mike Abbot, che con Matt Alexander (in basso) creò Hard Hat Mack. Al centro in basso John Field (Axis Assassin e The Last Gladiator). Infine, David Maynard, autore di WORMS.

Le nuove Maxell

UD I

Vanta una tecnologia senza confronti nella sua categoria. Infatti tutte le proprietà magnetiche del suo nastro sono esaltate al massimo grazie alla nuova particella magnetica "Ferricrystal" che, a differenza di quelle tradizionali, è **assolutamente non porosa**. La UD I, quindi, offre un livello d'uscita migliorato su tutta la gamma di frequenze, specialmente nelle medie e basse; una più ampia gamma dinamica, ed ottime caratteristiche di lownoise. Adotta la meccanica P.A. (PHASE ACCURACY) per garantire la massima stabilità di svolgimento del nastro e per contenere la differenza di fase tra i canali stereo entro i 10°. Può essere usata su qualsiasi tipo di registratore.

UD II

È la soluzione ideale per chi vuole una cassetta di **categoria superiore** ad un **costo contenuto**. Utilizza un nastro di posizione "Chromo" (CrO_2) prodotto con la stessa tecnologia dei nastri XL e XL-S Maxell. Infatti il nastro della UD II è composto dalle collaudate particelle magnetiche "Fine Epitaxial", ulteriormente perfezionate. Di conseguenza è aumentata tutta la **gamma dinamica** del nastro, il livello d'uscita alle **medie** ed **alte** frequenze, mentre il livello di rumore di bias ed il rumore di modulazione sono ridotti praticamente a **zero**.

Il meccanismo di scorrimento P.A. e la perfezione dei gusci della cassetta assicurano un regolare svolgimento del nastro, mantenendolo sempre perpendicolare alla testina del registratore (differenza di fase entro i 10°).

maxell®

È TUTTA UN'ALTRA MUSICA.

DELPI

di un computer non compatibile col servizio postale tibetano nè italiano), ma non propone nulla di diverso dal "servizio" che, periodicamente, offrono le varie riviste in edicola.

Rinunciate anche, per carità, a trasformare il salotto buono di casa vostra in accogliente alcova per innocentissimi incontri tra soci i quali, pur considerandosi martiri a tutti gli effetti di legge, diserteranno le future riunioni dopo l'amara esperienza dei musi duri dei vostri familiari che non avranno trovato altro da fare, nella prima ed unica riunione, che sbadigliare, chiedere l'ora, presentarsi in pigiama ed accusare martellanti emicranie.

Morale: o trovate un tetto sotto il quale incontrarvi oppure rinunciate a fondare il "partito" (o frequentate il club che avrà risolto per voi il problema...)

La sede ideale deve avere uno spazio sufficiente per ospitare almeno 30 persone (sedute) in caso di conferenze. La possibilità di utilizzo di servizi igienici, pur se non esclusivi del locale è di certo una comodità innegabile. Indispensabile è, invece, disporre di un contatore Enel assolutamente autonomo e non in comune con altri uffici o locali attigui.

Un locale di 40 metri quadri rappresenta, pertanto, il minimo indispensabile per fregiarlo del nome di club dopo aver tolto il cartello "Fittasi loculo" che era sulla porta d'ingresso. Scherzi a parte, l'affitto di un locale di dimensioni anche piccole, adibito ad uso ufficio, può costare un bel mucchio di soldi che, in un modo o in un altro, deve assolutamente rientrare nelle casse del club.

A volte un asciutto ed ampio garage può rappresentare degnamente la sede, a patto, naturalmente, che non risulti interrato e sia invece posto al livello del marciapiede di un tranquillo condominio. Una buona verniciata in colori chiari, la copertura in linoleum del pavimento, un semplice ed economico riscaldamento a bombole di gas, una finta parete in legno truciolare, con porta, all'apertura della saracinesca del garage è l'ideale per il gruppo di squattrinati studenti che non possono permettersi altra sede. Non dimentichiamo che il secondo personal computer della storia (l'Apple) è nato proprio in un garage californiano.

Un'idea da non sottovalutare, specialmente in provincia, è quella di affittare un vero ufficio in comproprietà con altri club. Non è infatti necessario che la sede sia aperta tutti i giorni, anche perchè non tutti hanno la possibilità di frequentarlo assiduamente. L'apertura a giorni alterni, viceversa, induce a concentrare in pochi giorni (addirittura ore) l'attività dello stesso club, conferendo l'immagine di elevata produttività e "serietà" che di certo non nuoce a qualsiasi associazione.

E' ovvio che le due (o tre) associazioni, in base a precisi accordi preliminari, usufruiranno dei locali nei giorni stabiliti: il lunedì ed il giovedì svolgerà la propria attività il Computer Club; il martedì e il venerdì toccherà al Club Elettronico; il mercoledì ed il sabato l'Associazione Rabdomanti si scambierà bastoncini e bicchieri d'acqua; la domenica si organizzano spedizioni punitive nei confronti di coloro che hanno lasciato in disordine i locali.

La coabitazione forzata impone, infatti, il rispetto reciproco tra i soci dei vari club. Ideale (e più economico di quanto si immagini) risulta in questi casi affidare l'ordine dei locali all'impresa di pulizia del condominio cui appartiene la sede.

Sarebbe preferibile che la sede di un Computer Club fosse utilizzata in time sharing con associazioni di hobbisti di elettronica, radioamatori, utenti di videoregistratori ed altre persone che potrebbero offrire la propria esperienza per collegare le varie attività, diverse solo in parte.

Vi sono, comunque, molte opportunità per usufruire di luoghi idonei, magari gratis. Sono numerosi ormai, specie in provincia, autorevoli personalità (presidi, direttori didattici, direttori di biblioteche comunali, assessori alla pubblica istruzione eccetera), sensibili al richiamo affascinante dell'informatica che metterebbero volentieri a disposizione di un serio gruppo di persone i locali di una scuola, della biblioteca pubblica o di idonei uffici comunali.

E' ovvio che, in questi casi, la correttezza dei partecipanti alle riunioni deve essere esemplare e tale da giustificare la loro presenza all'interno di un locale pubblico il cui scopo è, principalmente, quello di "produrre cultura".

Spesso, comunque, le difficoltà non provengono dalle massime autorità scolastiche o comunali: è a volte necessario, infatti, convincere, ricorrendo a stucchevoli lusinghe, soprattutto bidelli e custodi che vedono con sospetto la richiesta di prolungare l'orario di apertura dei locali presso i quali prestano servizio.

Non dimenticate, inoltre, che per motivi opposti e identici potrebbero offrire le proprie sedi dirigenti di partiti politici (che approfitterebbero per rifilarvi allarmanti reportage su situazioni politico sindacali) e parroci o responsabili di sette religiose di vario tipo che non perderebbero l'occasione per mettervi di fronte alle vostre responsabilità di peccatori.

Accettare una sede nei casi appena esaminati, potrebbe però conferire, al club, una fisionomia decisamente estranea alle finalità prefissate. Molto spesso, comunque, specie in provincia, utilizzare una particolare sede politica o religiosa, non costituisce necessariamante un "compromesso" o, peggio, un cedimento. In ogni caso, come direbbe Frà Cristoforo (cfr. Manzoni) "Omnia munda mundis" (tutto è innocente per coloro che sono innocenti) ed anche: "Ma chi se ne f...: gli affari sono affari".

L'arredo

Supponendo di dover condividere la sede con altre associazioni, sarà opportuno prevedere la sistemazione di un grosso contenitore (armadio) in cui custodire le apparecchiature più delicate durante la vostra assenza.

Un paio di vecchi televisori sistemati su di un lungo ed ampio tavolo costituiranno l'uscita video per i computer. Ogni socio, volendo, potrà attrezzarsi con una capiente borsa in cui riporre le apparecchiature che porterà con sè tutte le volte, nel caso in cui la fiducia nel ritrovare lo stesso numero di oggetti sia modesta.

Sono necessarie almeno quattro sedie oltre a un divanetto per tre posti (che fa tanto salotto fin de siecle), che può essere pietosamente contrabbandato per arredamento "giovane" ed essenziale. Nel caso in cui, al contrario, è possibile disporre di una maggior quantità di denaro, non sarà difficile arredare gli ambienti con mobili confortevoli e di gradevole effetto. Importante, infatti, è la prima impressione ricevuta dagli aspiranti soci che, prima di iscriversi, desiderano giustamente dare un'occhiata alla sede.

In ogni caso risulta indispensabile installare almeno una quindicina di prese di corrente oltre ad un efficace impianto di messa a terra: non scherzate con la sicurezza personale ed evitate assolutamente fili volanti come conduttori della tensione di rete.

Manifesti pubblicitari, poster, tabelle di codici macchina, schemi elettrici degli apparecchi ed altre pubblicazioni che riguardano da vicino il mondo dell'informatica risulteranno gradevoli e soprattutto utili in qualsiasi Computer Club.

I soci fondatori

Il gruppo che decide la fondazione del club deve assolutamemte essere affiatato allo scopo di evitare discussioni, naufragi di secolari amicizie e formazioni di faide. Non accettate chiunque solo perchè vi manca una quota di denaro o ritenete di non poter assolvere ai numerosi compiti. Meglio rinviare la costituzione del club o limitarsi ad una sede più modesta piuttosto che accettare una persona che, in seguito, potrebbe crear grane.

Allo scopo di evitare spiacevoli equivoci provvedete in ogni caso a mettere per iscritto le norme di conduzione del club, specialmente nei casi seguenti:

- diritto di occupazione della sede da parte dei soci o di iscritti durante le ferie ed i periodi in cui il club rimane chiuso;
- ripartizione e suddivisione degli oggetti e del denaro in cassa nel caso di defezione di un socio;
- impegno al pagamento della quota dell'Enel o della Sip per tutto il periodo di utilizzo della sede. Evitate comunque l'allacciamento telefonico oppure, se proprio lo reputate opportuno, richiedete il contascatti;
- ripartizione delle eventuali spese di riparazione degli apparecchi in uso comune;
- ripartizione degli impegni a tenere aperto il club durante l'orario stabilito;-massima chiarezza nel caso (frequente) in cui è possibile ricavare un utile nel caso di corsi di computer, stesura di programmi, consulenze di ogni tipo;
- divisione di responsabilità nel caso di esperimenti di elettronica che risultassero dannosi per le apparecchiature comuni o di un socio.

Le mansioni

Oltre a versare una quota per le prime spese (limitandosi a sperare che possa esser recuperata dalle iscrizioni e/o dai servizi che il club offrirà ai soci), è indispensabile che ogni socio abbia la responsabilità di uno specifico compito:

- il presidente provvede a indire gli incontri, a contattare eventuali sponsor e a tenere le pubbliche relazioni in generale. E' bene che questa persona sia la più anziana del gruppo, dotata di savoir faire e di carisma. Evitate di nominare presidente una persona troppo entusiasta, rigidamente tecnica, facile ai litigi e che disprezza tutto ciò che non sia al silicio. Scartate anche, per ovvie ragioni di parte, il commerciante di computer; pensate invece a un insegnante, un medico, un avvocato, un commerciante di fotoottica;
- il vicepresidente provvede a portare a termine i contatti iniziati dal presidente. Interviene in caso di difficoltà e sostituisce il presidente qualora sia assente per diversi motivi;
- il segretario si occupa dei contatti con i soci e riferisce al presidente. Se possibile, il suo è il numero di telefono ufficiale del club;
- il tesoriere è colui che sarà visto col massimo sospetto da ciascun iscritto. E' bene quindi che la contabilità sia tenuta correttamente fino all'ultima lira. Nel caso in cui siano in ballo grosse cifre, aprite un conto corrente presso una banca e chiedete che vi siano più intestatari del conto;
- il bibliotecario si occuperà degli abbonamenti alle riviste italiane e straniere, provvederà a segnalare gli articoli più importanti e terrà il libro dei volumi e fascicoli prestati (previa cauzione) ai soci che ne facciano richiesta. Terrà aggiornata anche la biblioteca del software annotando per ciascun programma su supporto magnetico il computer su cui gira, eventuali espansioni o accessori richiesti, eccetera;
- altri soci possono occuparsi dei rapporti col proprietario dell'immobile, con l'impresa delle pulizie, con i centri diriparazione e di altre mansioni che potessero utili.

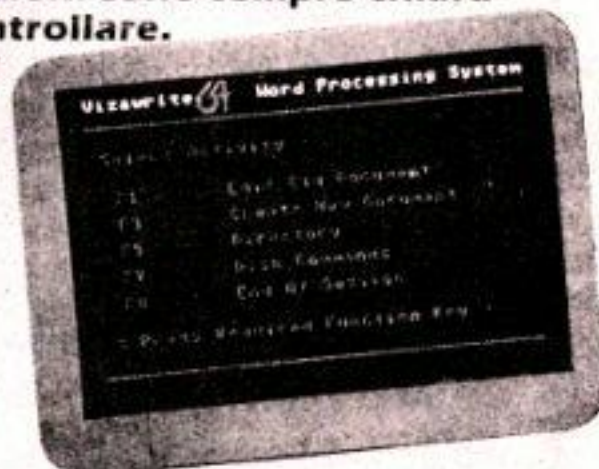
Il primo...

Torneremo sull'argomento, di certo interessante e richiesto da più parti. Nel frattempo rileggete attentamente queste righe e non abbiate fretta di aprire un club: dedicate il vostro tempo a individuare, tra i vostri amici e conoscenti, le persone più idonee a svolgere le varie mansioni. Scartate dall'elenco, senza pietà, persone che, pur se amabilissime, non possono sostenere determinati ruoli.

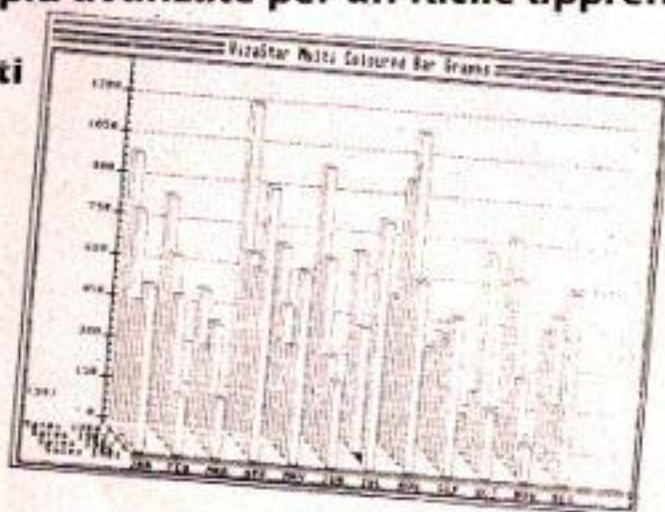
Nel prossimo numero, poi, una sorpresa piacevole per i più tenaci: Commodore Computer Club (e la SYSTEMS Editoriale al completo) vi verrà incontro con una serie di iniziative che....

VIZAWRITE

IL PERSONAL WORD PROCESSOR PIÙ COMPLETO ED EVOLUTO.

Vizawrite è un word processor ad alte prestazioni studiato appositamente per il Commodore 64. Vizawrite sfrutta al massimo i colori, la grafica e l'ampiezza di memoria del tuo computer, per offrirti le caratteristiche e la classe di un word processor professionale: formatta il testo istantaneamente mentre lo inserisci; la linea di formato controlla margini e tabulazioni, scorre totalmente il documento; ricerca e sostituisce frasi in modo selettivo; imposta automaticamente il margine sinistro; sottolinea, esalta in grassetto, copia, sposta e cancella qualsiasi parte del testo. Inoltre le sue numerose funzioni sono sempre chiaramente indicate (mediante speciali simboli grafici) e facili da controllare.
Vizawrite è il word processor ideale per gli utenti più esigenti.
VIZAWRITE È DISPONIBILE ANCHE PER IBM PC.

VIZASTAR

IL PROCESSORE DI INFORMAZIONE

Vizastar è l'unico programma nel suo genere per il Commodore 64. Totalmente compatibile con il word processor Vizawrite, insieme costituiscono un completo sistema per l'ufficio.
Vizastar ha integrati gli importanti aiuti elettronici per la tua attività sia di lavoro che a casa (un foglio elettronico, un database e grafica in alta risoluzione video).
Viene fornito con uno speciale menù per una pratica selezione delle operazioni-opzioni.
Estrae dai tuoi archivi le informazioni e le include nel foglio di lavoro disegnandole istantaneamente sullo schermo, per calcoli veloci e potenti.
La dimensione del foglio elettronico è di 1.000 righe per 64 colonne, ha una grande capacità di memoria e con l'inserimento di speciali «finestre» permette di velare una sezione del foglio di lavoro mentre si lavora su di un'altra.
Vizastar è stato realizzato sfruttando le tecniche più avanzate per un facile apprendimento ed una immediata familiarità dell'utente.
Adesso puoi gestire le tue informazioni importanti in tanti modi diversi con un metodo veloce ed efficace.

Distribuiti in Italia da:

EASY COMPUTING®

Via A.Bertani 24 - 50137 Firenze

PRELUDIO MUSICALE

Cos'è un suono? Cosa è una frequenza? come si stabiliscono le pause di uno spartito? E, soprattutto: come la teoria musicale può entrare nel computer fino a generare la musica? Ecco un breve prontuario per orientarsi tra le sette note elettroniche.

Tutti conosciamo quella forma d'espressione chiamata: MUSICA. Essa come altri arti quali la pittura, la poesia, la scultura, ha un suo mezzo di espressione: il SUONO.

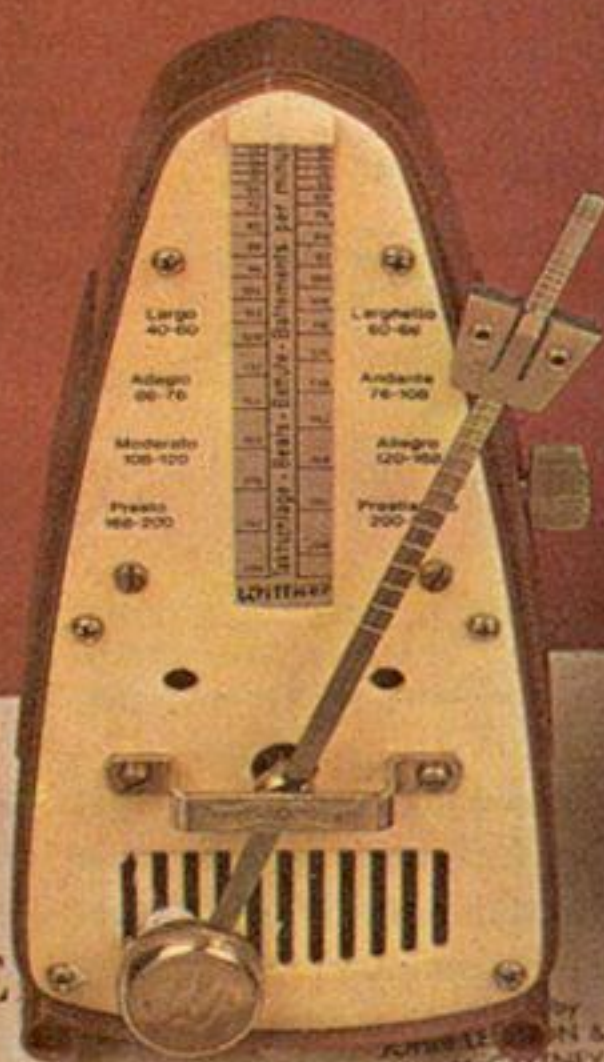
Ma che cos'è il suono e qual è la sua causa? E' una sensazione che avvertiamo grazie al nostro orecchio: la voce di una persona, il rumore di un motore, le note emesse da un pianoforte, il cinguettio di un uccello, sono tutti dei suoni. Ma cerchiamo di capirne la sua fisiologia facendo un esperimento: prendiamo una lamina metallica lunga circa 20-30 centimetri e poggiamola sul bordo di un tavolo facendola di almeno metà della sua lunghezza. Mantenendo ferma con una mano l'estremità poggiata sul tavolo (figura 1), con l'altra diamo un colpetto all'estremità sporgente. Udiremo un suono e noteremo che la lamina compie dei movimenti che vanno dal basso verso l'alto e viceversa.

Questi movimenti si chiamano VIBRAZIONI. Quando finiscono non non si sentirà più alcun suono. Possiamo affermare che il suono è provocato dalle vibrazioni di un corpo elastico. Ma come fanno a giungere al nostro orecchio, o meglio, come si propaga il suono?

Facciamo un altro esperimento lanciando un sasso in uno stagno. Osserveremo che dal punto in cui cade, si originano onde concentriche circolari sempre più grandi (figura 2). Si potrebbe pensare, dunque, che si sia verificato uno spostamento d'acqua. Ripetiamo allora, l'esperienza con delle modifiche.

Prendiamo una bacinella colma d'acqua, e mettiamo un piccolo pezzo di sughero. Lasciando cadere una monetina noteremo che si formano onde concentriche sempre più grandi come nell'esperimento precedente.

Osserviamo, però, che sono proprio le onde a spostarsi dal punto in cui è caduta la monetina fino ai bordi della bacinella, e non l'acqua: se fosse quest'ultima a muoversi trascinerebbe il sughero verso i bordi della bacinella. Questa esperienza dimostra che le molecole dell'acqua non si spostano nella direzione in cui l'onda si propaga, ma oscillano intorno a posizioni di equilibrio, nelle quali si trovavano prima della sollecitazione.

Il suono, dunque, si propaga per mezzo di onde sonore. Se volessimo rappresentare graficamente un'onda sonora, otterremmo il disegno di figura 3. La curva rappresentata in tale grafico ha il nome di sinusoide. Il tratto di onda che va dal punto A al punto B si chiama ciclo. Il tempo impiegato per compiere tale ciclo si chiama periodo. La quantità di cicli compiuti in un secondo si chiama frequenza e si misura in hertz ossia cicli compiuti in un secondo. Se diciamo che

un suono ha la frequeza di 200 Hertz, significa che compie in un secondo 200 cicli. Nel caso del primo esperimento, ciò vuol dire che la lamina vibra (cioè va sù e giù) 200 volte in un secondo.

Tre proprietà fondamentali

Possiamo stabilire fra tre proprietà del suono. La prima è l'altezza, dovuta alla frequenza. Se è bassa avremo suoni gravi (o bassi); se è alta avremo suoni acuti (o alti)

Per stabilire la seconda proprietà riprendiamo il primo esperimento. Dopo aver colpito leggermente l'estremita libera della lamina ed averne osservato l'effetto sonoro, proviamo a colpirla più energicamente. Noteremo che il suono emesso è sempre lo stesso, ma nel primo caso avevamo ottenuto un suono debole, mentre nel secondo uno forte. E' come se una persona dicesse una parola prima sottovoce poi urlando. Questo efetto determina l'intensità, proprietà del suono che permette di distinguere i suoni in "forti" e "deboli".

Figura 2: onde provocate dalla caduta di un sasso in uno stagno.

Figura 1: una lamina colpita compie vibrazioni.

Continuiamo, ora, il discorso sulla frequenza. Si è detto che un suono è definito da una frequenza. Ma ciò non basta. In effetti, ogni suono ha una frequenza chiamata fondamentale e altre frequenze, multiple e sottomultiple di questa chaimate armoniche. E' come se in una sala con, ad esempio, dieci persone, nove bisbigliassero ed una, invece urlasse. Noi che ci troviamo accanto, sentiremo maggiormente la voce di colui che grida.

Dunque, la persona la persona che urla, rappresenta la frequenza fondamentale di un suono. Le rimanenti nove rappresentano le armoniche.

Come nella sala vi sono dieci persone anche se ne udiamo una, così nel suono vi sono più frequenze anche se ne distinguiamo una sola, la fondamentale. Le armoniche sono molto importanti, la loro quantità e qualità determinano la terza proprietà del suono: il timbro. Ed è proprio grazie a quest'ultimo che distinguiamo il suono emesso da un pianoforte da quello di una chitarra o di un violino.

Osserviamo la figura 3. Rappresenta graficamente, un'onda sonora.

Più precisamente, rappresenta la frequenza fondamentale. Le figure 4 e 5 sono invece due armoniche di un'onda sonora. Unendo queste tre rappresentazioni, (sommando cioè i valori ipotetici delle tre frequenze) otterremo il grafico di figura 6, che rappresenta l'onda sonora composta dalla frequenza fondamentale e da due armoniche superiori.

Per meglio comprendere le fasi di un suono, effettuiamo un altro semplice esperimento.

Battiamo un tasto di pianoforte e teniamolo premuto. Noteremo che si ottiene un suono forte, che poi si abbassa e dura fino a quando terremo il tasto premuto. Difatti, se alzeremo il dito, il suono scenderà a volume zero.

Quattro fasi di suono

Deduciamo, dunque, che qualsiasi emissione sonora è caratterizzata da fasi. Con precisione sono quattro e si chiamano: Attack, Decay, Sustain, Release.

L'attack (attacco), è il tempo in cui il suono raggiunge il volume massimo.

Il decay (decadenza), è il tempo in cui il suono si porta dal volume massimo al volume "normale".

Il sustain (sostegno-durata), è il tempo in cui il suono continua a sentirsi.

Infine il release (rilascio), è il tempo in cui il suono dal volume "normale" scende a zero.

La figura 7 rappresenta graficamente le 4 fasi del suono. Ogni fase è caratterizzata da valori numerici. Più sono grandi tanto più durerà la fase definita.

Ad esempio, dire che l'attack ha valore uguale a 80 significa che il suono dal volume zero arriverà al volume massimo

Figura 3: rappresentazione grafica di un'onda sinusoidale. Il tratto A-B rappresenta un ciclo.

in 56 millisecondi. Dire invece che l'attack ha valore 208, vuol dire che il suono da zero arriverà al volume massimo in 2 secondi circa. E' logico che se l'attack è uguale a zero si avrà solo un rumore secco: il tempo impiegato dal suono, che da volume zero sale a volume massimo, è talmente piccolo (2 millesecondi), che il nostro orecchio non riesce a percepire la rapida varazione di volume.

Nella tabella 1 sono riportati, per alcuni valori delle fasi, le relative durate nel tempo.

Le quattro fasi vengono attivate dal generatore di inviluppo. Che cos'è?

Figura 4: rappresentazione grafica della prima armonica superiore.

Facciamo un esempio. Battendo un tamburo, notiamo che il suono sale rapidamente al volume massimo per poi rapidamente scendere a zero. Come si nota dalla figura 8, che rappresenta tale andamento, si ha un attack velocissimo ed un decay-release altrettanto veloce (difatti le linee sono quasi verticali). Il sustain, invece, è quasi nullo. Se così non fosse, dopo aver battuto il tamburo, il suono avrebbe dovuto continuare a sentirsi.

Prendiamo in esame un altro strumento: il pianoforte. Battiamo un tasto e teniamolo premuto ascoltandone, come al solito, la sua emissione sonora. La figura 9 rappresenta l'andamento del suono. L'attack sale al volume massimo, meno velocemente di prima. Si ha la fase di decay, sustain e release. Da notare che, questa volta, le ultime due fasi sono molte lunghe.

Difatti il suono emesso dal pianoforte, prima di giungere a volume zero, è durato un tempo maggiore di quello ottenuto dall'esperienza precedente. Tutti i fenomeni fisici finora descritti, concorrono alla formazione o alterazione di un suono.

Il sintetizzatore è uno strumento musicale che sintetizza una frequenza affinchè questo strumento emetta un suono, bisogna definire tutti i parametri per ottenere il suono.

Il Commodore 64 e la musica

Una delle cose veramente eccezionali che il Commodore 64 può fare è suonare.

Infatti il nostro "piccolo grande amico" ha all'interno un circuito integrato speciale chiamato SID, che è un vero e proprio sintetizzatore musicale. Inizia in $D400 (decimale 54272) ed occupa 29 locazioni di memoria fino a $D41C (dec.54300). Ognuna di queste 29 locazioni costituisce un "registro" del SID. In ognuno di essi dobbiamo memorizzare dati che definiscono i parametri del suono da ottenere. L'istruzione è la seguente: POKE numero registro, dato numerico.

La tabella 2 riporta tali registri e le loro funzioni.

La tabella 1 indica invece il tempo di durata della fase di attack o di decay (in millisecondi o in secondi) che si può ottenere usando i valori numerici corrispondenti al tempo prescelto.

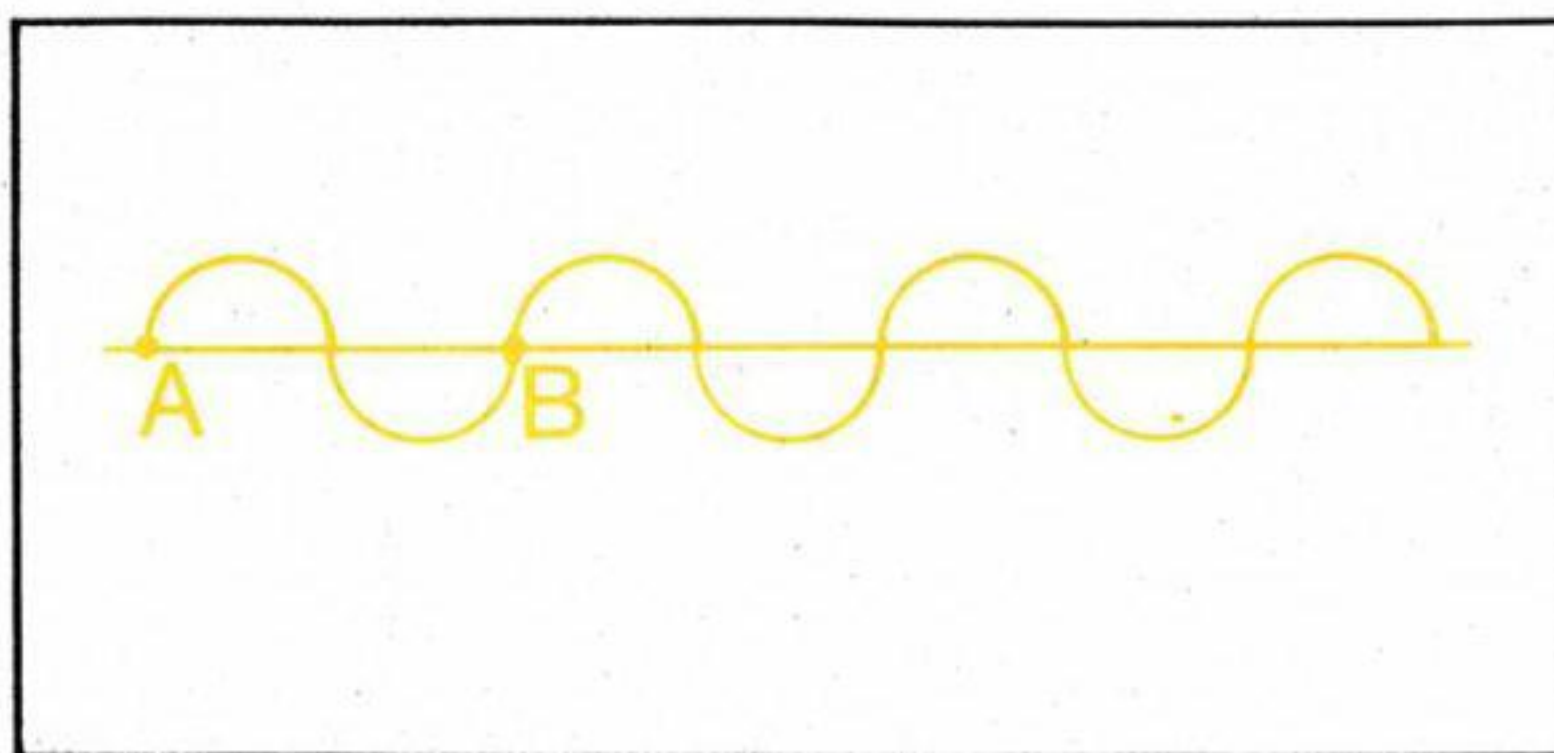
Per i meno esperti spieghiamo da dove saltano fuori i valori numerici riportati, ed anche il significato di "nibble alto" e di "nibble basso".

Innanzitutto bisogna dire che le informazioni su valore di attack decay occupano un solo byte, cioè una cella di memoria composta da otto bit.

Per poter introdurre le due informazioni (attack decay) nello stesso byte si è diviso (teoricamente) il byte in due parti, ciascuna di 4 bit.

La prima parte, che è composta da bit 4,5,6,7 si chiama nibble alto. In questo modo è come se nello stesso registro ci fossero due sottoregistri, ognuno dei quali ha a disposizione 4 bit.

Ricordiamo che mentre in 8 bit posso-

no essere memorizzati numeri compresi fra 0 e 255, in 4 bit possono essere memorizzati numeri compresi fra 0 e 15. Questo spiega il fatto che i valori numerici di attack e decay possono variare fra 0 e 15.

Una volta decisi i valori di attack e di decay da usare (che come spiegato devono essere compresi fra 0 e 15), bisogna considerare che il valore di attack va messo nel nibble alto del byte, per cui bisogna "shiftare" il numero (che, essendo compreso fra 0 e 15 appartiene al nibble basso), di 4 bit a sinistra.

Questo in parole povere vuol dire moltiplicare per 16 il valore numerico dell'attack.

Il valore ottenuto va sommato al valore dei decay ed è questo il numero (che se non avete commesso errori sarà compreso fra 0 e 255) da pokkare nel registro di attack/decay. Facciamo un esempio: scegliendo come tempo di attack 500 millisecondi.

Osservando la tabella si vede che il valore numerico dell'attack (di durata 500 millisecondi) è 10 (numero fra parentesi). Dobbiamo ora moltiplicare questo valore per 16, per ottenere il valore del nibble alto: il risultato è 160.

Figura 6: rappresentazione grafica della somma della frequenza fondamentale, prima e seconda armonica.

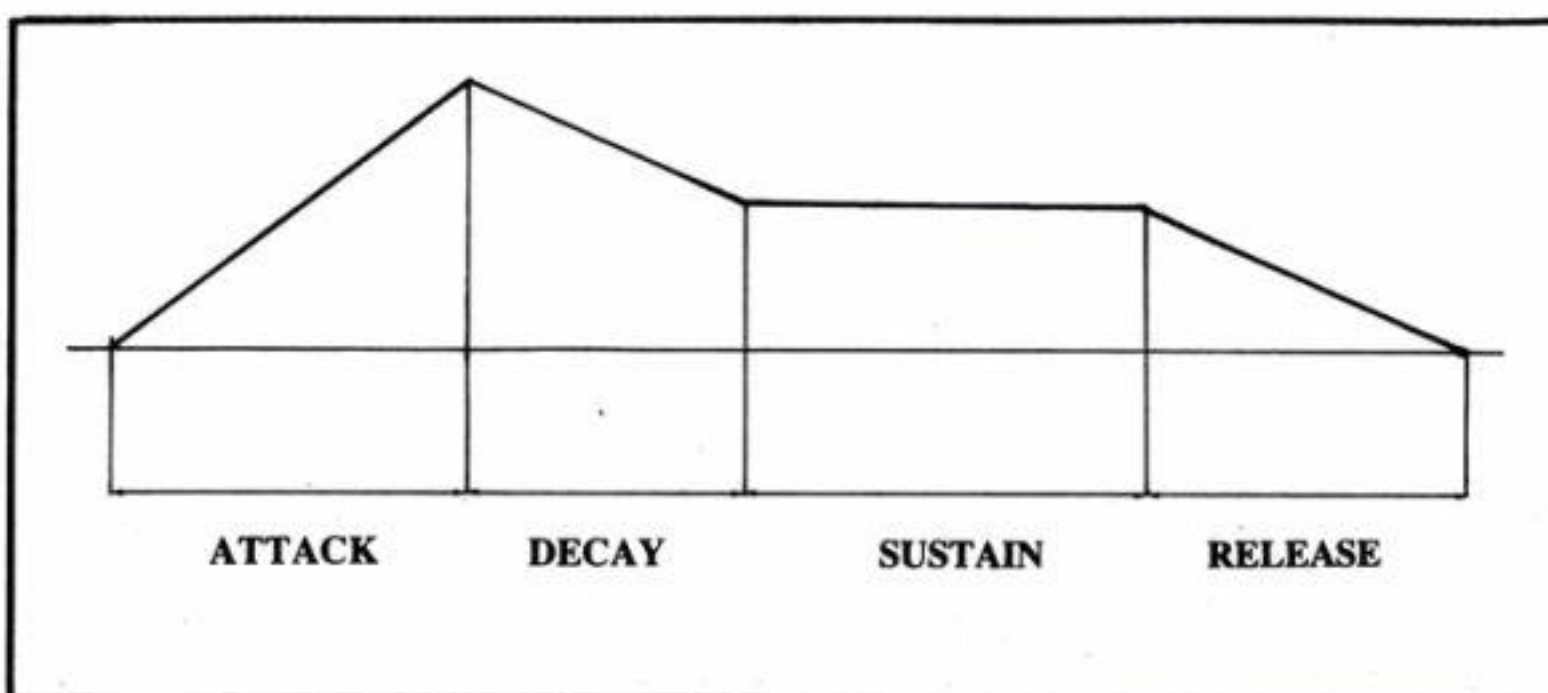



Figura 7: inviluppo del suono.

Figura 5: rappresentazione grafica della seconda armonica superiore.

Volendo esaminare i BIT di ogni operazione eseguita, si ha che se il valore di attack è uguale a 10 (500 millisecondi) = 00001010 (il numero è tutto nel nibble basso) moltiplicando per 16 si ottiene 160 = 10100000 (il numero è tutto nel nibble alto). E il valore decay è uguale a 13 (9 secondi) = 00001101 (il numero è tutto nel nibble basso).

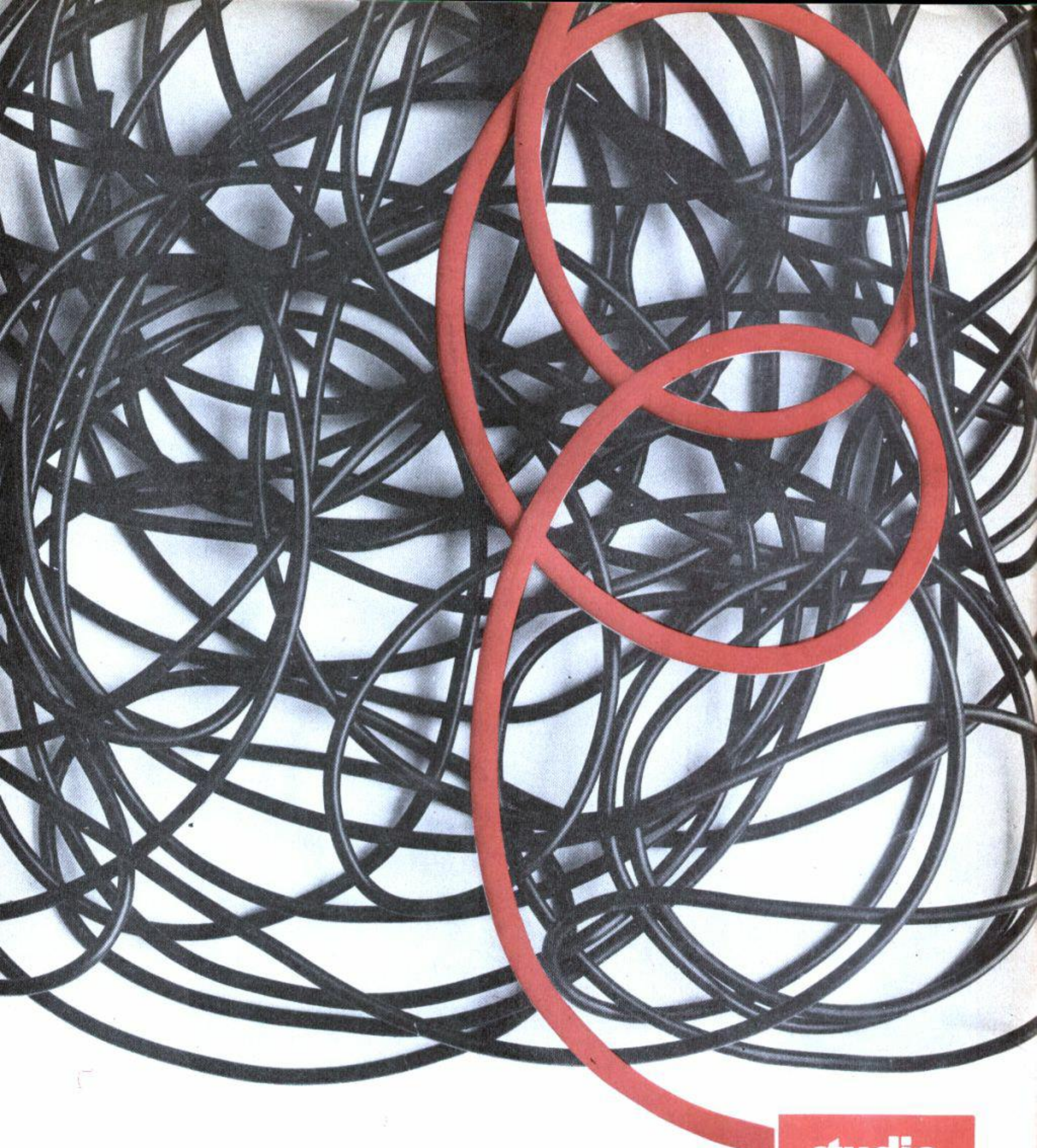
La somma di attack+decay risulta 10101101 = 173 in decimale.

Le stesse considerazioni si possono fare per i valori di sustain e release (uno nel nibble alto e uno nel nibble basso). Faccio notare che i tempi di decay corrispondono (per lo stesso valore) ai tempi di release.

Per non fare la moltiplicazione basta vedere subito il valore del nibble alto indicato nella tabella. Supponiamo, poi, che vogliamo un tempo di decay di 9 secondi: nella tabella, a questo valore di tempo corrisponde il numero 13, che è il nibble basso (quindi non va moltiplicato per 16 ma lasciato inalterato, per sapere quale valore pokkare nella locazione di attack/decay, dobbiamo fare la somma dei due numeri ottenuti: 160+13=173.

Pokando quindi 173, si avrà un attack di 500 millisecondi ed un decay di 9.

Dovrebbero essere chiari, a questo punto, i concetti sui quattro parametri fondamentali che caratterizzano ogni suono: riproducibile col C 64. Vedremo ora in che modo è possibile programmare correttamente il SID, il circuito inte-

grato del Commodore 64 che gestisce, appunto, il suono.

Il SID del 64

Il programma pubblicato in queste pagine consente di effettuare alcuni esperimenti. Non solo le peculiarità del SID ma anche, e soprattutto, per evitare alcuni errori molto comuni, specie tra i principianti. Quanto segue, tuttavia, è solo un accenno sull'impiego che si può fare del SID.

C'è da dire, innanzitutto, che per programmare correttamente il SID è necessario attenersi alle seguenti regolette:

- Regolare il volume
- Imporre l'attack-decay
- Imporre il sustain-release
- Selezionare (hi-lo) la nota musicale

A questo punto delle operazioni il suono è pronto per "scattare". Non appena sarà selezionata l'onda musicale, il programma sonoro andrà in esecuzione. Supponendo di voler utilizzare l'onda triangolare, sarà necessario settare il quinto bit del registro in oggetto. Tale valore è quindi 2↑5=32

Contemporaneamente, però, è necessario comunicare l'intenzione di sentire il suono. Come? settando il bit zero dello stesso byte. Il valore visto prima diventa, pertanto:
2↑5+1=33 (onda triangolare).

In modo perfettamente analogo si possono selezionare le altre onde (es.65=sinusoidale). Mettere al valore unitario il bit zero rappresenta, in altre parole, dare il RUN alla sintesi sonora.

Figura 8: inviluppo del suono emesso da un tamburo.

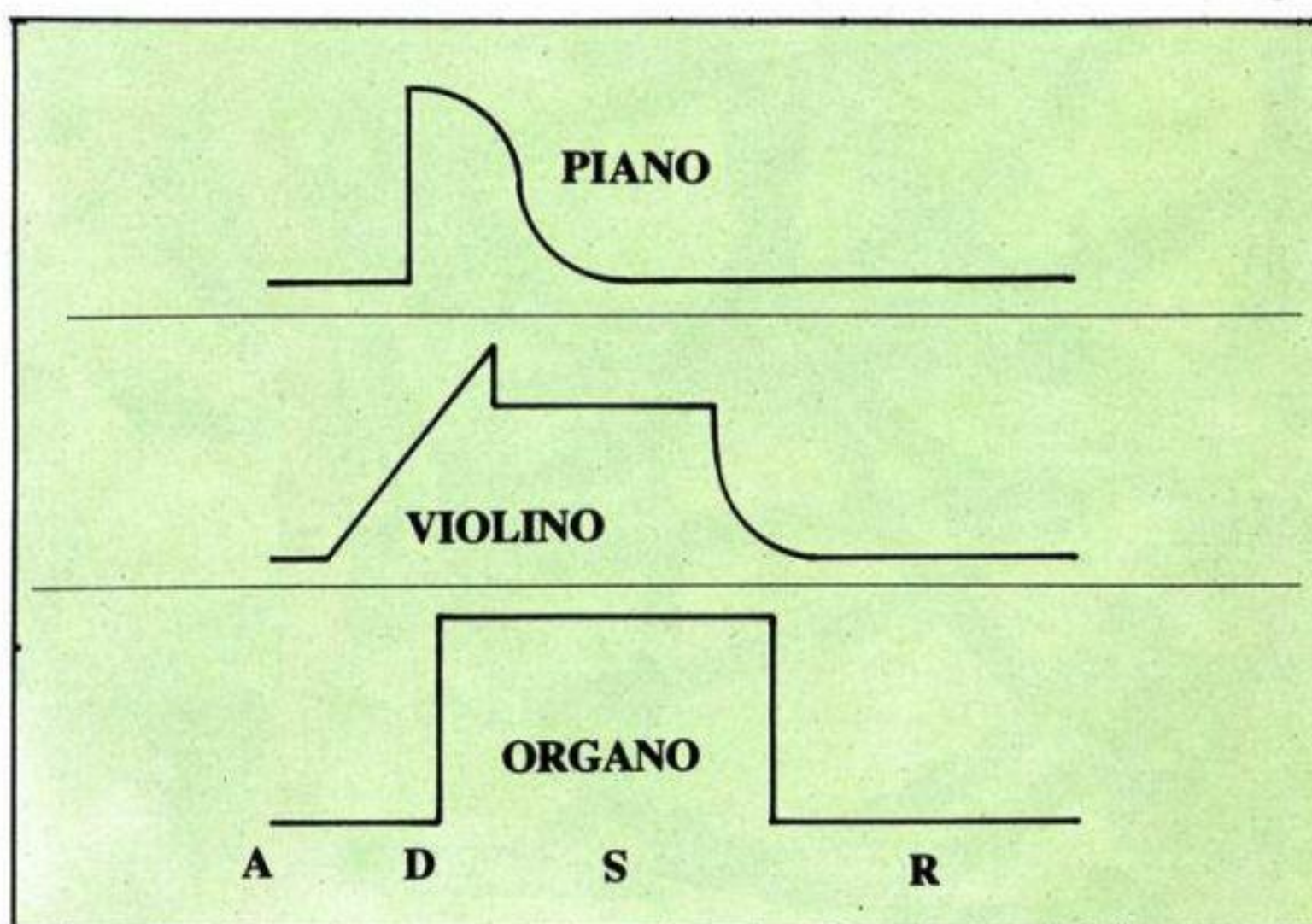



Figura 9: inviluppo del suono emesso da strumenti musicali.

Non appena si "poka" il valore 33 nel registro di selezione sonora si udrà il suono partire dal valore nullo fino al valore massimo consentito dal volume (fase di attack). In seguito il volume, nel realizzare la fase di decay, diminuisce fino al valore indicato dal sustain. A questo punto il suono prosegue all'infinito.

Spieghiamoci meglio: se il livello imposto col sustain è massimo (15), la fase di decay, in pratica, non si realizza. Se, al contrario, il sustain viene posto ad un valore inferiore (esempio:8) si attuerà dapprima la fase di attack, poi il decay si incaricherà di diminuire, nel tempo programmato, il volume sonoro fino a stabilizzarlo sul valore determinato dal nibble alto della poke sustain.

Conclusioni

Quando il bit zero del byte che determina l'onda viene posto a zero, imzierà la fase del release (decadimento).

In definitiva:

- L'attack, il decay e il release determinano la durata del tempo impiegato dal suono per raggiungere il massimo livello (attack), per diminuirlo (decay) fino al livello del sustain (a patto che questo lo permetta) e per rilasciarlo (release) dal livello del sustain fino al volume nullo.
- Il sustain determina il livello sonoro cui la nota selezionata deve restare indefinitivamente fino a che il bit zero del byte di selezione onda resterà alto.
- Per udire nuovamente la nota musicale non sarà necessario riprogrammare il tutto, ma basterà ripristinare il valore nel byte di selezione d'onda.

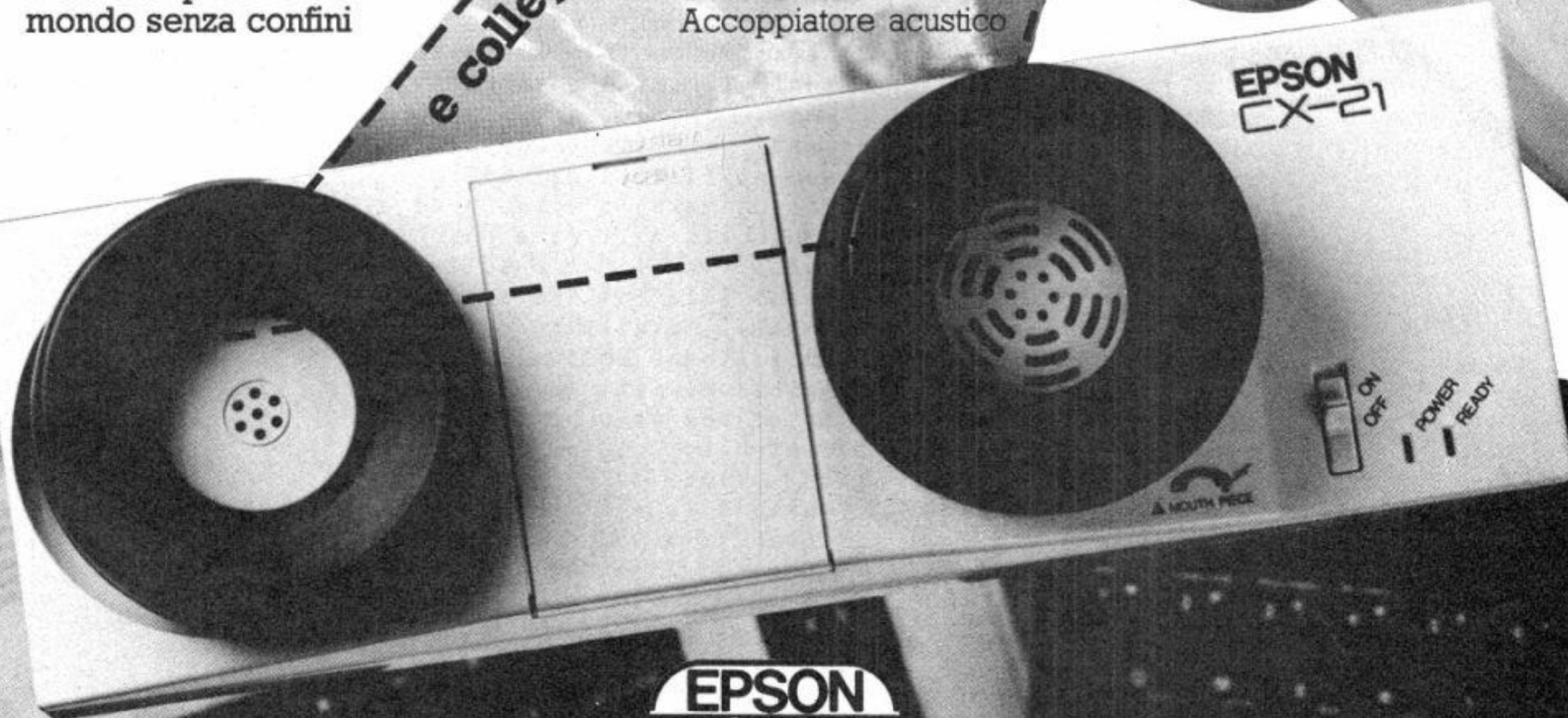
COMPUTERTELEFONA
I TUOI
PROGRAMMI
CX-21 EPSON è un accessorio che dà un valore enorme al tuo calcolatore, quello dei terminali telecomunicazione.
CX-21 è un modem costruito con le tecnologie più avanzate, (modulatore - demodulatore) capace di convertire i segnali digitali in suoni che possono essere trasmessi con un normale telefono, in tutta sicurezza.
Con CX-21 puoi:
– "telefonare" i programmi direttamente al computer dei tuoi amici che posseggono un CX-21
– comunicare con personal computer di altre marche
– collegarti con le banche dati di tutto il mondo
CX-21 è progettato per adattarsi ai diversi apparecchi telefonici per consentire lo scambio veloce e facile di programmi e dati. CX-21 è costruito con speciali circuiti integrati CMOS, è piccolo, leggero e molto affidabile.
Caratteristiche
– Funzionamento a batterie ricaricabili (4 ore di autonomia)
– Funzionamento duplex o half duplex
– Velocità di trasmissione fino a 300 bit-sec.
– Interfaccia RS 232
– Dimensioni 297x95x42 mm
– Peso 700 gr.
Inoltre CX-21 può essere collegato a qualsiasi home o personal computer tramite interfaccia RS 232 - C.
e collegati alle banche dati di tutto il mondo
EPSON•SEGI Garanzia delle Quattro Stagioni
EPSON CX-21, il tuo computer in un mondo senza confini
EPSON CX-21
Accoppiatore acustico
EPSON CX-21
ON OFF
POWER READY
MOUTH PIECE
EPSON SEGI
EPSON - SECI S.p.A. - Milano - Via Timavo 12 - Tel. 02/6709136
studio martinetti

54272	0	Frequenza nota byte basso
54273	1	Frequenza nota Byte alto
54274	2	Ampiezza onda quadra byte basso
54275	3	Ampiezza onda quadra byte alto
54276	4	Registro contr ollo forma d'onda
bit 0		on=abilita le forme d'onda
bit 1		on=sincronizza l'oscillatore 1 e 3
bit 2		on=abilita l'o scillatore 1 e 3
bit 3		on=disabilita l'oscillatore 1
bit 4		on=abilita l'o nda triangolare
bit 5		on=abilita l'o nda a dente di sega
bit 6		on=abilita l'o nda quadra
bit 7		on=abilita il rumore bianco
54277	5	Registro contr ollo Attack-Decay
bit 7,6,5,4		definizione at tack
bit 3,2,1,0		definizione de cay
54278	6	Registro contr ollo Sustain-Release
bit 7,6,5,4		definizione su stain
bit 3,2,1,0		definizione re lease
54279	7	Frequenza note byte basso
54280	8	Frequenza note byte alto
54281	9	Ampiezza onda quadra byte basso
54282	10	Ampiezza onda quadra byte alto
54283	11	Registro contr ollo forma d'onda analogia col r egistro 4
54284	12	Registro contr ollo Attack-Decay analogia col r egistro 5
54285	13	Registro contr ollo Sustain-Release analogia col r egistro 6
54286	14	Frequenza nota byte basso
54287	15	Frequenza nota byte alto
54288	16	Registro contr ollo forma d'onda analogia col r egistro 4
54289	17	Ampiezza onda quadra byte basso
54290	18	Ampiezza onda quadra byte alto
54291	19	Registro contr ollo Attack-Decay analogia col r egistro 5
54292	20	Registro contr ollo Sustain-Release analogia col r egistro 6
54293	21	Frequenza di t aglio byte basso
bit 0,1,2		usati per tale frequenza
bit 3,4,5,6,7		non utilizzati dal SID
54294	22	Frequenza di t aglio byte alto
54295	23	Controllo filt ri e risonanza
bit 0		on=abilita fil tro sulla voce 1
bit 1		on=abilita fil tro sulla voce 2
bit 2		on=abilita fil tro sulla voce 3
bit 3		on=filtra even tuali ingressi audio
bit 4,5,6,7		selezione filt ro di risonanza
54296	24	Selezione volu me e filtri
bit 0,1,2,3		controllo volu me sonoro
bit 4		on=abilita fil tro Passa-Basso
bit 5		on=abilita fil tro Passa-Banda
bit 6		on=abilita fil tro Passa-Alto
bit 7		on=disabilita la voce 3
54297	25	Lettura dati X paddle
54298	26	Lettura dati Y paddle
54299	27	Lettura oscill atore voce 3
54300	28	Lettura ADSR d ella voce 3

TABELLA 1

ATTACK		+	DECAY	
VALORE NUMERICO (Nibble alto)	TEMPO DI ATTACK	+	VALORE NUMERICO (Nibble basso)	TEMPO DI DECAY (e RELEASE)
0 (0)	2 Millisecondi	+	0	6 Millisecondi
16 (1)	8 "	+	1	24 "
32 (2)	16 "	+	2	48 "
48 (3)	24 "	+	3	72 "
64 (4)	38 "	+	4	114 "
80 (5)	56 "	+	5	168 "
96 (6)	68 "	+	6	204 "
112 (7)	80 "	+	7	240 "
128 (8)	100 "	+	8	300 "
144 (9)	250 "	+	9	750 "
160 (10)	500 "	+	10	1.5 Secondi
176 (11)	800 "	+	11	2.4 "
192 (12)	1 Secondo	+	12	3 "
208 (13)	2 Secondi	+	13	9 "
224 (14)	5 "	+	14	15 "
240 (15)	8 "	+	15	24 "

```
100 REM  *** COMMODORE 64: INTRODU
    ZIONE ALLA MUSICA ***
110 :
120 S=54272: REM  INDIRIZZO INIZIL
    E SID
130 FOR  T=0  TO  24: POKE  S+T,0:
     NEXT : REM  AZZERAMENTO
140 :
150 INPUT "[CLEAR]ATTACK (0-15)";A
    K
160 AK=AK*16: REM  VALORE DI ATTAC
    K
170 INPUT "DECAY (0-15)";DY
180 INPUT "SUSTAIN (0-15)";SN
190 SN=SN*16: REM  VALORE DI SUSTA
    IN
200 INPUT "TEMPO DI SUSTAIN (X/60
    SEC.)";T
210 INPUT  "RELEASE (0-15)";RE
220 :
230 PRINT"[CLEAR]ATTACK"AK/16"DECA
    Y"DY" [RVS]TOTALE"AK+DY
240 PRINT"SUSTAIN"SN/16"RELEASE"RE
    " [RVS]TOTALE"SN+RE
250 PRINT T/60" SECONDI"
260 POKE  S+24,15  : REM  VOLUME
270 POKE  S+5,AK+DY: REM  ATTACK/D
    ECAY
280 POKE  S+6,SN+RE: REM  SUSTAIN/
    RELEASE
290 REM   POKE S,49: POKE S+1,28:
    REM NOTA "LA"
300 POKE  S,49: POKE  S+1,28: REM
     NOTA "LA"
310 POKE  S+4,17:    REM  ONDA TRI
    ANGOLARE
320 TI$="000000"
330 PRINT"[HOME]"  TAB(32) TI$: IF
      TI<T  THEN  330
340 POKE  S+4,16: REM  SOLO PER ON
    DA TRIANGOLARE
350 IF  H  THEN  400
360 PRINT"[3 DOWN][RVS]CHE COSA VU
    OI FARE?"
370 PRINT"[DOWN]1/ RIPETERE L'IMPO
    STAZIONE"
380 PRINT"[DOWN]2/ CAMBIARE IMPOST
    AZIONE"
390 PRINT"[DOWN]3/ FINE LAVORO"
400 GET A$:IF A$="" THEN 400
410 H=1:IF A$="1" THEN 310
420 IF A$="2" THEN  RUN
430 IF  A$="3"  THEN POKE S+24,0:
    END : REM  VOLUME A ZERO
440 GOTO   400
```

TEMPO DI METRONOMO

di **Stefano Tulli**

Un breve programma utile anche a coloro che desiderano studiare la programmazione di ritmi musicali.

Questo programma, scritto parte in BASIC, parte in linguaggio macchina, realizza un metronomo e può pertanto essere utile agli studenti di musica.

La linea 100 prepara lo schermo e le locazioni del SID, vale a dire del circuito integrato che gestisce l'attività sonora del computer. Il carattere "N" in reverse si ottiene digitando CTRL-N, e serve per passare dal modo maiuscolo-semigrafico al modo maiuscolo-minuscolo.

Viene poi caricata (READ... DATA) la routine in L.M. nelle locazioni del buffer del nastro e, se avete sbagliato nel digitare i valori dei DATA, viene segnalato l'errore. Il programma chiede poi il tempo desiderato (che deve esser compreso tra due e dodici movimenti), quindi la velocità in quarti (battiti) al minuto, che è la notazione usata abitualmente per i metronomi. In seguito tale valore viene convertito in sessantesimi di secondo e memorizzato nelle locazioni decimali 0825 e 0827.

A questo punto esegue la routine L.M. posta a partire da 033C (esadecimale), che ha solo il compito di modificare il vettore di IRQ (Interrupt Request) e azzerare la locazione $a2, cioè il byte basso dell'orologio del C-64. Da questo punto viene eseguita la nuova routine di IRQ, posta a $035A, che confronta la locazione $a2 con $0827 e, nel caso risultino eguali, la azzera ed emette un suono, più acuto di un'ottava all'inizio di ogni battuta.

Contemporaneamente il programma BASIC attende (istruzione WAIT) la pressione del tasto F1, dopodichè ristabilisce il vettore originale di IRQ ($ EA31) spegnendo così il metronomo e ricominciando nuovamente il ciclo. Sicuramente questo programma non soddisferà il gusto "estetico" di tutti i lettori, ma la tecnica utilizzata, con le eventuali personalizzazioni, permetterà di eseguire musica con una temporizzazione migliore di quella ottenibile mediante consueti cicli d'attesa FOR...NEXT.

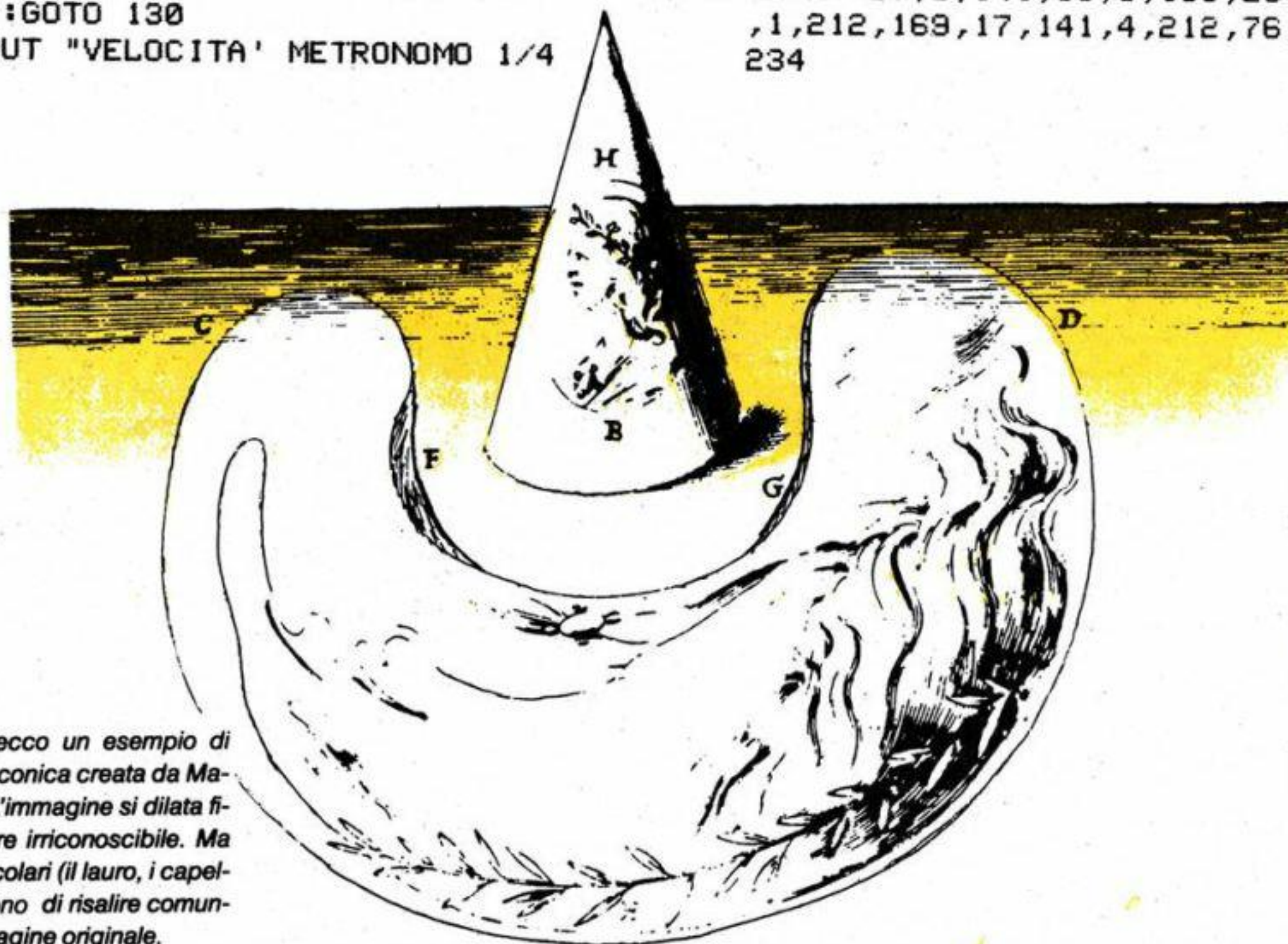
E' comunque doveroso precisare che con tempi di metronomo maggiori di 300 il timer del 64 mostra i suoi limiti. Il bello del programma pubblicato consiste nel fatto che funziona contemporaneamente ad altri programmi BASIC.

Se, infatti, interrompete il programma (tasto RUN/STOP) durante il suo funzionamento, vi accorgerete che il metrono-

mo continua a funzionare anche mentre listate, caricate altri programmi da disco eccetera.

E' bene comunque ricordare che l'uso del registratore a cassette cancella la routine L.M. dato che questa è "ospitata" proprio nelle locazioni su cui agiscono le routine del nastro.

Chi vuol saperne di più sulla modifica del vettore di Interrupt può rileggere l'articolo apparso su C.C.C. N. 13 di settembre 84.

```
 10 REM  COMMODORE 64
 30 REM  METRONOMO
 35 :
 40 REM  STEFANO TULLI
 90 :
 99 REM  SCHERMO E SUONO
100 S=54272:POKE S+24,15:POKE S+1,
    20
105 POKE S+5,5:PRINT"[BIANCO][MIN-
    MAI]"
109 REM  *** CARICAMENTO ROUTINE L
    .M. **
110 FOR I=828 TO 900:READ A:POKE I
    ,A:CK=CK+A:NEXT:IF CK<>7343 TH
    EN PRINT"ERRORE":END
119 REM  **** INPUT VELOCITA' ****
120 PRINT"[CLEAR] METRONOMO"
130 INPUT "CHE TEMPO VUOI(DA 2 A 1
    2)";T
132 IF T<2 OR T>12 THEN PRINT"[2 U
    P]":GOTO 130
135 INPUT "VELOCITA' METRONOMO 1/4
    = ";MM$
140 MM=VAL(LEFT$(MM$,3))
145 IF LEN(MM$)>3 OR MM<30 OR MM>2
    50 THEN PRINT"[2 UP]":GOTO 135
149 REM  **** START ****
150 PM=3600/MM:POKE 825,T:POKE 826
    ,1:POKE 827,PM:SYS828
159 REM  **** FINE  ****
160 PRINT"PREMI F1 PER FINIRE"
170 WAIT  197,4: SYS 845:GOTO 120
179 REM  **** ROUTINE L.M. ****
180 DATA  120,169,90,141,20,3,169,
    3,141,21,3,169,0,133,162,88,96
190 DATA  120,169,49,141,20,3,169,
    234,141,21,3,88,96,165,162,205
    ,59,3,208,33
200 DATA  169,0,133,162,141,4,212,
    169,16,141,1,212,206,58,3,208,
    11,173
210 DATA  57,3,141,58,3,169,20,141
    ,1,212,169,17,141,4,212,76,49,
    234
```

Nel 1642, ecco un esempio di anamorfosi conica creata da Mario Bettini. L'immagine si dilata fino a risultare irriconoscibile. Ma alcuni particolari (il lauro, i capelli) permettono di risalire comunque all'immagine originale.

IL GIOCO DEL RIMBALZO

Come muovere uno sprite in una direzione voluta e farlo rimbalzare quando incontra un ostacolo.

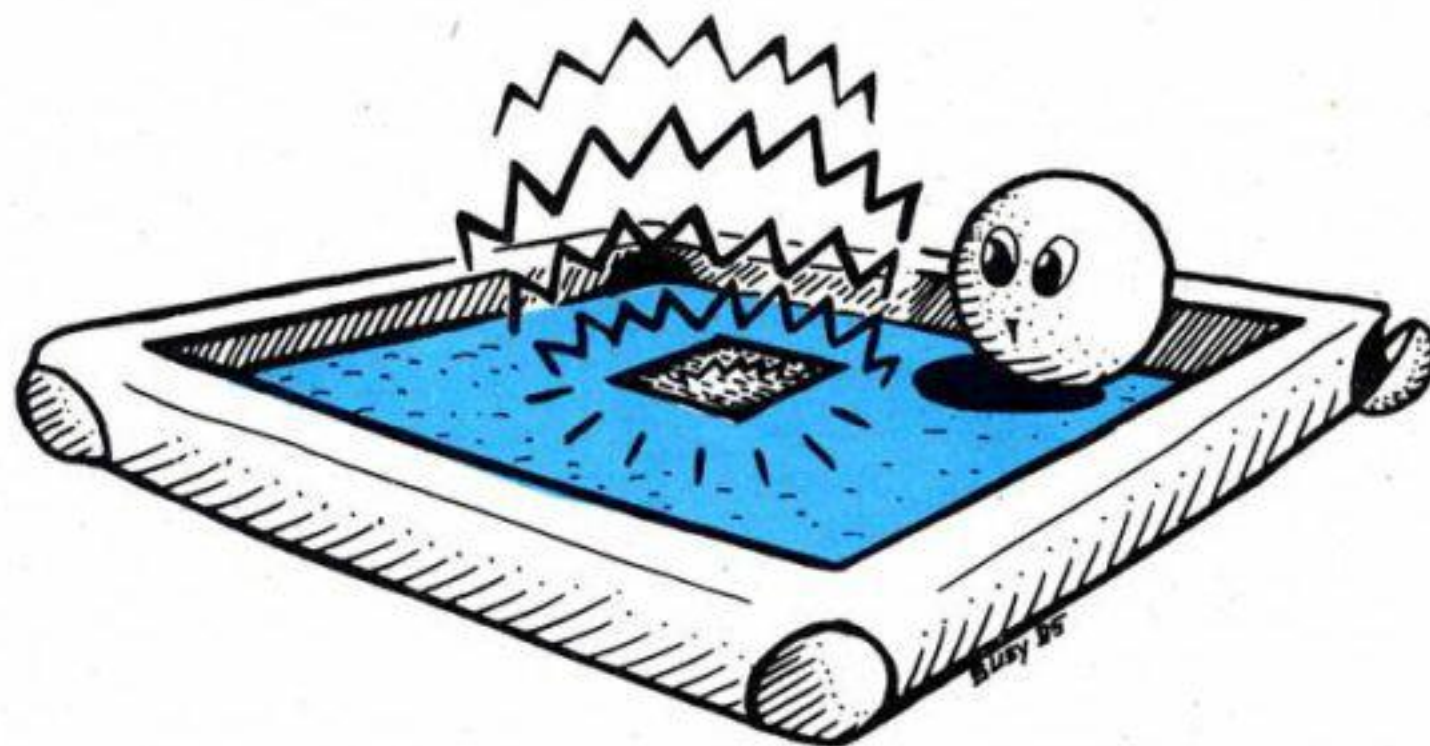
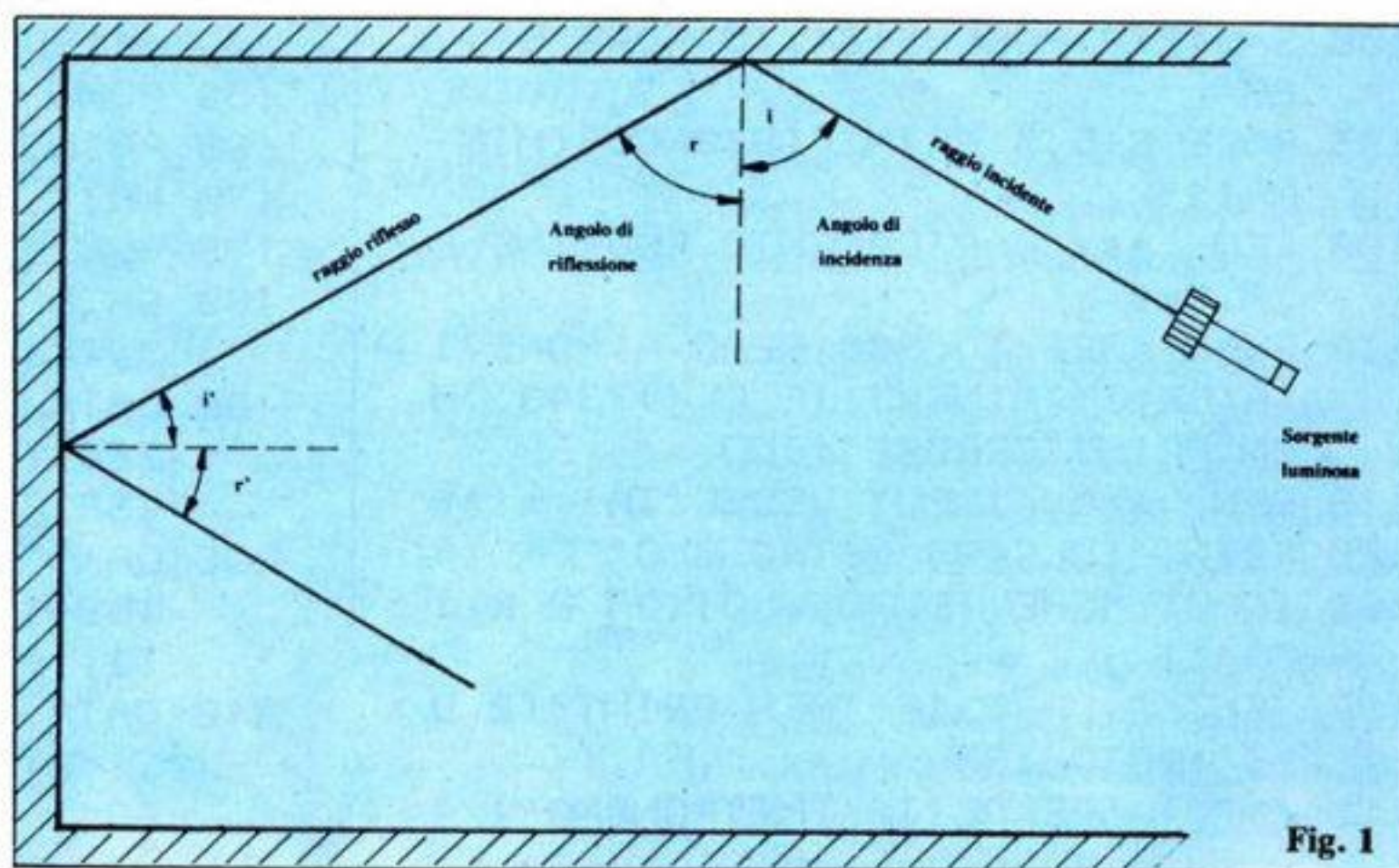
Una pallina è posta entro un campo rettangolare. Bisogna colpire una o più volte (al massimo 4) le sponde e centrare il bersaglio (quadrato rosso) che è situato nel campo. Il gioco assomiglia vagamente a quello del biliardo e infatti le traiettorie della palla seguono gli stessi principi. Per determinare la direzione si deve battere in INPUT l'angolo di tiro, è meno comodo che avere in mano una stecca ma, se siete abili, dopo qualche prova acquisterete l'occhio e la precisione sufficienti per padroneggiare le geometrie del gioco.

Una volta dato il RUN, verrà chiesto il livello di difficoltà determinato dalla presenza di ostacoli più o meno numerosi che ci intralceranno durante la partita.

Per ogni tiro dovrete poi battere l'angolo (in gradi, da -90 a +90) e premere <RETURN>. Nella linea 200 la variabile N indica il numero di tiri per partita; SM è invece il massimo numero di sponde (meno una) che è consentito colpire.

Fig. 1

Un po' di teoria

Per la programmazione di video giochi del tipo biliardo, carambola o che contengono comunque effetti quali proiettili che rimbalzano, raggi laser riflessi, eccetera è utile conoscere qualche nozione sulla "riflessione"
cosa del resto abbastanza intuitiva.

Da notare che al secondo rimbalzo l'angolo incidente non varrà più "i", ma sarà l'angolo ad esso complementare:

i' = pigreco/2-i (radianti)
o anche:
i' = 90-i (gradi)

e così via proseguendo nel percorso.

Guardiamo la figura 1: una sorgente luminosa proietta un raggio di luce contro uno specchio che, dopo la riflessione, formerà due angoli, tra i quali vale la seguente relazione:

angolo (i) d'incidenza = angolo (r) di riflessione

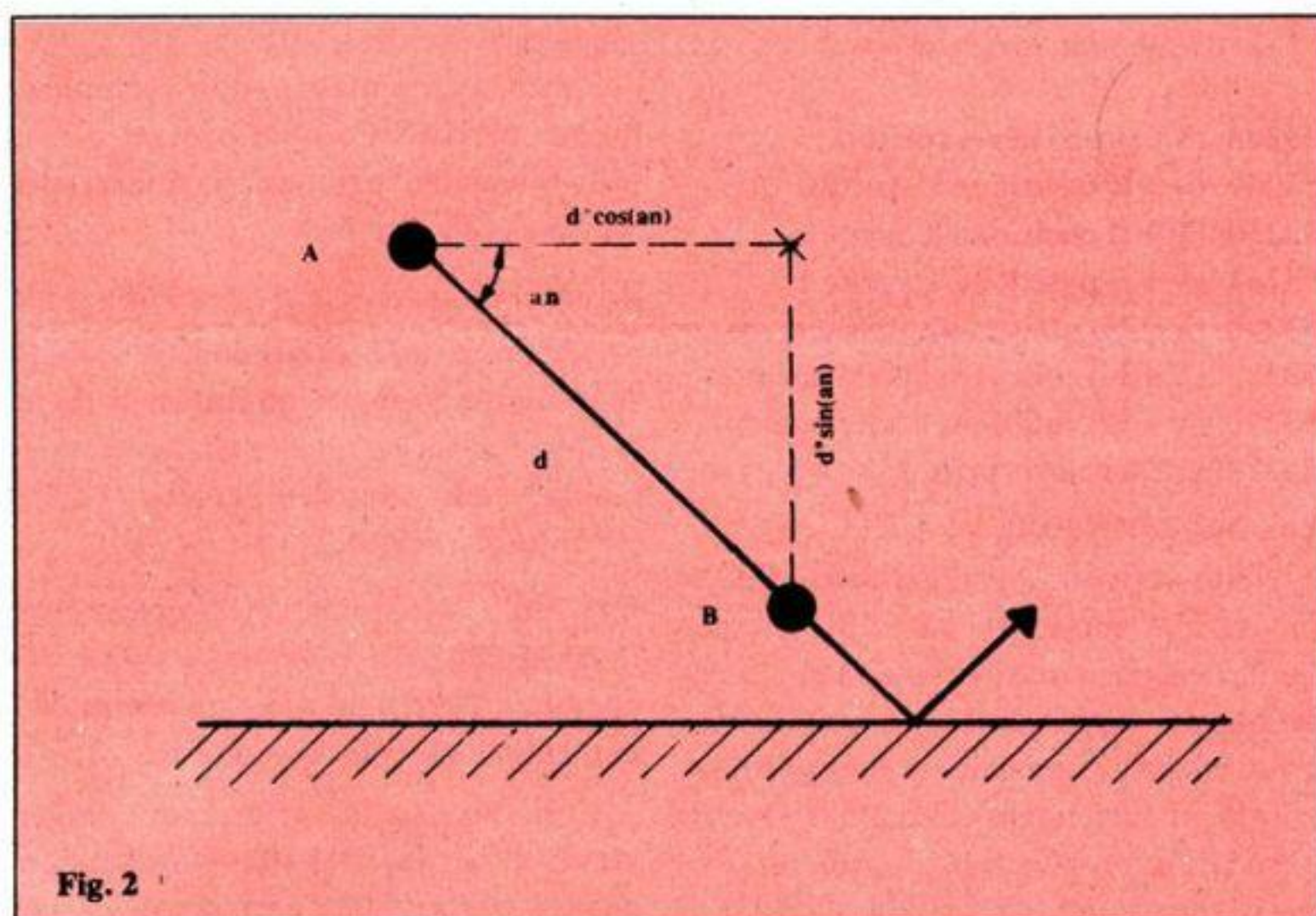



Fig. 2

In pratica il problema si può presentare come in fig.2: uno sprite deve essere mosso da un punto "A" e passare per il punto "B", per poi proseguire fino a che non troverà un ostacolo o una sponda (casi, questi, che vedremo più avanti). Detta "d" la distanza fra A e B, e "an" l'angolo della traiettoria, le coordinate dello sprite dopo lo spostamento saranno:

X = X + d*COS(an) (coord. orizzontale)
Y = Y = d*SIN(an) (coord. verticale)

Queste due relazioni permettono di sapere in ogni istante la posizione del nostro oggetto. Bisogna ricordare che sul video del C-64 gli assi cartesiani non hanno la disposizione convenzionale: il punto di coordinate 0,0 si trova in alto a sinistra. Chi non ha molta familiarità con le funzioni trigonometriche non si spaventi, questa è la prima e unica volta che compariranno, le espressioni usate saranno sempre le stesse.

Finora sappiamo come agire per spostare uno sprite da un punto all'altro; il passo successivo è quello di dare l'impressione del movimento, spostandolo di un pixel (puntino luminoso elementare) per volta; cioè la distanza "d" deve essere rapportata all'unità di misura del nostro video (il pixel per l'appunto). In parole povere, ponendo d=1 avremo:

X = X + COS(an)
Y = Y + SIN(an)

Le variabili X e Y vengono immesse nei rispettivi registri per il movimento:

```
POKE 53248,X:POKE 53249,Y
```

se usiamo ad esempio lo sprite numero 0.

Infine, con un semplice loop, i valori X ed Y vengono di volta in volta incrementati e lo sprite si muove nella direzione voluta.

Il rimbalzo

Per far cambiare la direzione allo sprite (palla, proiettile o altro) secondo la legge della riflessione, bisogna anzitutto sapere quando arriverà nella posizione critica. Nel programma le sponde sono rappresentate dal bordo dello schermo, perciò, dopo aver mosso la pallina, bisogna controllare se questa si trova in corrispondenza di uno dei bordi:

```
720 IF X< 21 OR X> 249 THEN CO=-CO...
```

Se la coordinata X è minore di 21 o maggiore di 249 (estremi del campo), allora il movimento orizzontale deve essere invertito.

"CO" vale COS(a), come visto in precedenza. Per invertire la traiettoria basta quindi cambiare semplicemente il segno a CO.

```
730 IF Y< 48 or Y> 230 THEN SI=-SI...
```

...inverte il movimento verticale. Vale lo stesso discorso fatto in precedenza.

Collisioni

Sempre in questo gioco sono previsti due tipi di collisioni:

***1/* Collisione sprite-sfondo.**

Accade ogni volta che la palla si scontra con uno dei caratteri messi nel campo di gioco. Per verificare se ciò è avvenuto, bisogna leggere il contenuto del r egistro 53279. Questo byte controlla le collisioni con lo sfondo di tutti gli sprite utilizzati (max 8) e, essendo ogni byte composto di 8 bit, ognuno di questi controlla uno sprite (per una spiegazione più dettagliata potete leggere l'articolo Studio Sprite sul numero 18 di C.C.C.).

```
770 IF PEEK(V3)=2 THEN 800
```

Se la locazione 53279 è posta a 1, allora si eseguono le operazioni relative alla collisione sprite-sfondo.

***2/* Collisione sprite-sprite.**

Per incrementare il punteggio occorre colpire lo sprite rettangolare ottenuto con la riga 280. Anche qui c'è un solo registro (53278) per verificare le possibili collisioni di due o più sprite fra loro. Quando due o più bit di questa locazione di memoria valgono "1" allora siamo nella situazione accennata.

```
780 IF PEEK(V2)=3 THEN 810
```

Se il registro 53278=3, allora si è fatto centro.

Da dove si ricava questo 3? Dato che in partenza abbiamo assegnato alla pallina lo sprite 0 e al bersaglio lo sprite 1, per

sapere se collidono il valore letto deve risultare:

2 + 2 = 1 + 2 = 3

Se, ad esempio, i rispettivi sprite fossero stati il 5 e il 6, sarebbe stato così:

2 + 2 = 32 + 64 = 96

Per concludere

Da quanto detto finora avrete capito che queste particolari locazioni, con informazioni così "concentrate", sono abbastanza fastidiose da manipolare, anche se non presentano grossissime difficoltà concettuali. Quindi prestate le dovute a ttenzioni, sempre necessarie quando si ha a che fare con istruzioni PEEK e POKE, al fine di evitare effetti indesiderati, non escluso quello dell'inchiodamento del computer.

Riepiloghiamo i registri usati:

53248 (V) posizione X sprite 0
53249 (V+1) posizione Y sprite 0
53250 (V+2) posizione X sprite 1
53251 (V+3) posizione Y sprite 1
53269 (V+21) abilita, disabilita sprite
53278 (V+30) collisione sprite-sprite
53279 (V+31) collisione sprite-sfondo
53278-53294 colori sprite

Un po' più veloci

Nello scrivere un gioco in BASIC si incontrano molto spesso difficoltà per quello che riguarda la rapidità dei movimenti; nei video-games, dove la velocità è un fattore essenziale, si ricorre solitamente al linguaggio macchina. Questo però presenta alcuni inconvenienti, prima di tutto perchè si perde la comodità di programmazione tipica del BASIC e in secondo luogo perchè si propongono listati pieni di DATA, incomprensibili alla maggior parte dei lettori.

Il listato presentato è scritto completamente in BASIC; questo è stato reso possibile dall'adozione di alcuni accorgimenti che illustro brevemente:

650 SI=3*SIN(AN):CO=3*COS(AN)

Gli incrementi orizzontali e verticali sono moltiplicati per un fattore 3. Si sacrifica così un po' della pulizia nel movimento, ma in compenso la pallina viaggia tre volte più veloce.

690-760. Con le istruzioni di queste righe il controllo della posizione rispetto alla sponda è fatto ogni due movimenti ed il controllo delle collisioni ogni sei.

Così facendo lo sprite andrà leggermente a "strappi" (l'effetto è comunque appena percettibile), però ora la velocità è più che accettabile.

Flavio Molinari

```
100 REM  COMMODORE 64
110 :
120 REM  RIMBALZI
130 :
140 REM  BY FLAVIO MOLINARI
150 :
160 REM  *** INIZIALIZZAIONE VARIA
    BILI ***
170 POKE  53280,9:POKE  53281,5:PO
    KE  646,0:REM  COLORI SCHERMO
180 DIM  DN$(24):DN$(1)=CHR$(19)
190 FOR  Q=2  TO  24:DN$(Q)=CHR$(1
    9)+DN$(Q-1)+CHR$(17):NEXT
200 V=53248:V1=V+1:V2=V+30:V3=V+31
    :N=10:SM=5
210 FOR I=1 TO 7:C1$=C1$+CHR$(157)
    :NEXT
220 C1$=CHR$(18)+"       "+C1$:REM
     7 SPAZI
230 :
240 REM  *** SPRITE ***
250 POKE 2040,13:POKE  2041,14
260 POKE  V+39,7:POKE  V+40,2:REM
     COLORI SPRITE
270 FOR  K=0  TO  62:READ  A:POKE
     832+K,A:NEXT
280 FOR  K=0  TO  62:POKE  896+K,2
    55:NEXT
290 DATA  0,0,0,0,0,0,0,0,0,0,0,0,
    0,0,0,0
300 DATA  60,0,0,255,0,1,255,128,1
    ,255,128,3,255,192,3,255
310 DATA  192,3,255,192,3,255,192,
    1,255,128,1,255,128,0,255,0
320 DATA  0,60,0,0,0,0,0,0,0,0,0,0
    ,0,0,0
330 :
340 REM  *** SCELTA LIVELLO ***
350 POKE  V+21,0:PRINT CHR$(147)SP
    C(252)"LIVELLO (1-5)";
360 INPUT  L:IF  L<1  OR  L>5  THE
    N  350
370 NC=4*(L-1):REM  NUMERO CARATTE
    RI CASUALI
380 :
390 REM  *** CAMPO DI GIOCO ***
400 POKE  V+30,0:POKE  V+31,0
410 PRINT CHR$(147);:P2=0:SP=0:X=0
    :Y=0
420 FOR  Q=0  TO  24:POKE  1055+Q*
    40,160:NEXT
430 PRINT DN$(2 ) TAB(33)"PUNTI"
```

```
440 PRINT DN$(17) TAB(33)"TIRO"
450 PRINT DN$(20) TAB(33)"ANGOLO"
460 PRINT DN$(21) TAB(33)"-90 +90"
470 :
480 REM  *** CARATTERI CASUALI ***
490 IF  NC=0  THEN  530
500 FOR  K=1  TO  NC
510 X=INT(RND(1)*24)+5:Y=INT(RND(1
    )*23)+2
520 PRINT DN$(Y) TAB(X)CHR$(209):S
    O=1:GOSUB  950:NEXT
530 X=INT(RND(1)*80)+80:Y=INT(RND(
    1)*30)+80
540 POKE  V+2,X:POKE  V+3,Y:REM  P
    OSIZ. BERSAGLIO
550 POKE  V+21,2:FOR  Q=1  TO  20:
    NEXT:IF  PEEK(V3)>0  THEN  530
560 :
570 FOR  W=1  TO  N:PRINT DN$(17)
    TAB(37)W
580 X=25:Y=INT(RND(1)*180)+50:REM
     POSIZIONE INIZIALE PALLA
590 POKE  V,X:POKE  V1,Y:POKE  V+2
    1,3:REM  ABILITA SPRITE
600 :
610 REM  *** INPUT ANGOLO DI TIRO
    ***
620 PRINT DN$(23) TAB(33)C1$;
630 INPUT  AN$:AN=VAL(AN$):IF  AN<
    -90  OR  AN>90  THEN  620
640 AN=-π*AN/180:REM  CONVERSIONE
    IN RADIANTI
650 :SI=3*SIN(AN):CO=3*COS(AN)
660 :
670 REM  *** MOVIMENTO PALLA ***
680 SP=0:POKE  V3,0:POKE  V2,0
690 FOR  J=1  TO  3
700 X=X+CO:Y=Y+SI:POKE  V,X:POKE
    V1,Y
710 X=X+CO:Y=Y+SI:POKE  V,X:POKE
    V1,Y
720 IF  X<21  OR X>249  THEN  CO=-
    CO:GOTO  760
730 IF  Y<48  OR Y>230  THEN  SI=-
    SI:GOTO  760
740 NEXT
750 GOTO  770
760 SP=SP+1:SO=1:GOSUB  950:IF  SP
    =SM  THEN  800
770 IF  PEEK(V3)=1  THEN  800
780 IF  PEEK(V2)=3  THEN  810
790 GOTO  690
800 P1=0:SO=2:GOSUB  950:GOTO  830
810 IF  SP=0  THEN  800
820 P1=SP*400*L*L:SO=1:GOSUB  950:
    REM  PUNTI
830 FOR  Q=1  TO  2000:NEXT
840 :
850 REM  *** AGGIORNAMENTO PUNTEGG
    IO ***
860 P2=P2+P1
870 PRINT DN$(3) TAB(33)P2:NEXT W
880 :
890 REM  *** CONTINUAZIONE GIOCO *
    **
900 PRINT DN$(24)CHR$(18)" VUOI CO
    NTINUARE ? (S/N)"
910 GET  Q$:IF  Q$="N"  THEN  END
920 IF  Q$="S"  THEN  350
930 GOTO  910
940 :
950 REM  *** SUONO ***
960 POKE  54296,15:POKE  54277,10:
    POKE  54278,10
970 IF  SO=1  THEN  OD=17:FR=50:DU
    =30
980 IF  SO=2  THEN  OD=129:FR=8 :D
    U=500
990 POKE  54276,OD:POKE  54273,FR
995 FOR  Q=1  TO  DU:NEXT:POKE  54
    276,0:RETURN
```

9 CARTE

di **Flavio Molinari**

Fra "umani" il gioco si svolgerebbe più o meno così: si prendono, da un mazzo di carte francesi, nove carte dello stesso seme, dall'asso al nove. I valori sono distribuiti in questo modo: l'asso vale 1, il due vale 2 ... il nove 9. Si dispongono scoperte sul tavolo in modo che siano visibili. Due contendenti si alternano scegliendo una carta per volta. Vince chi per primo riesce ad arrivare esattamente a 15, sommando i valori di tre carte qualsiasi tra quelle scelte.

Un esempio vale più di mille spiegazioni.

Immaginiamo una partita tra i giocatori A e B. Parte per primo B e sceglie il 5. A prende il 7. B risponde prendendo l'8.

A questo punto A, per stoppare il gioco dell'avversario, deve scegliere il 2.

La situazione dopo la quarta mano è la seguente:

B:5,8

A:7,2

Tocca a B e sceglie il 6 per le stesse ragioni.

A prende il 4.

B prende l'1 e vince.

Situazione finale:

B 5,8,6,1 vince con 8+6+1=15

A 7,2,4

Se, dopo aver preso tutte le carte, nessuno dei due riesce a fare 15, la partita finisce in parità.

Un semplice gioco di strategia. Attenzione però... le insidie sono molteplici e battere il computer non è molto facile.

"Umani" contro computer

Il gioco può sembrare abbastanza banale, ma in realtà contiene parecchi tranelli ed è facile venire sconfitti da chi conosce già qualche "astuzia" (e il nostro C64 le conosce!).

Quando ritenete di essere diventati sufficientememte abili, potete cimentarvi nel livello 2, ma vi avverto: ora batterlo sarà estremamente difficile...

Due quesiti

Il gioco appena descritto è del tipo a "conoscenza completa"; infatti non esistono elementi casuali ed ognuno può vedere e pensare le proprie mosse. Con queste premesse sarebbe possibile analizzare a tavolino tutte le situazioni che si possono verificare durante una partita e trovare la strategia migliore.

Appartengono a questa classe di giochi anche gli scacchi, la dama ecc. Questi ultimi però hanno una complessità enormemente superiore.

Propongo ora due problemi:

1/ Se giocata perfettamente da entrambi le parti (gioc.-gioc. oppure gioc.-computer) la partita come terminerà?

Le possibilità possono essere tre:

a) vince chi fa la prima mossa;

b) vince chi fa la seconda mossa;

c) finisce in parità.

2/ Questo gioco, non molto noto, è viceversa conosciutissimo sotto un'altra veste. Quest'ultimo ha un aspetto completamente diverso , ma la strategia è assolutamente identica. Siete capaci di scoprire di cosa stiamo parlando? (sicuramente tutti noi abbiamo fatto una partita di...)

Se riuscite a scoprirlo saprete probabilmente dare una risposta anche alla prima domanda. Per darvi un aiuto dirò che bisogna prendere in considerazione i quadrati magici... ma non posso dirvi di più.

Strategie del computer

I problemi che si sono presentati nella programmazione sono evidentemente diversi da quelli dei video-games tradizionali. Infatti qui il computer deve svolgere anche una parte attiva per poter competere contro un giocatore umano e non semplicemente mostrare sprite cui sparacchiare, o muovere ometti su e giù per lo schermo.

Nel nostro caso deve fare una serie di "ragionamenti" prima di fare la sua scelta:

1/ vedere se alla prossima mossa può vincere prendendo una carta opportuna;
2/ altrimenti deve chiudere la strada all'avversario se questo ha la possibilità di vincere subito;
3/ se i primi due test hanno esito negativo, deve scegliere in maniera "giusta" per proseguire al meglio la partita.

Vediamo punto per punto come sono stati risolti i problemi nel programma.

1/ Giocata vincente? (linee 2000-2060)

Vengono analizzate tutte le possibili combinazioni delle carte in suo possesso. Se la ricerca ha esito negativo prosegue al punto 2.

2/ Giocata forzata? (2200-2250)

Stesso procedimento adottato in precedenza, analizzando però le carte dell'avversario.

3/ Strategie del computer (2400-2490)

Sono suddivise in due parti: la prima è quella di tipo generale ed è composta di sole tre righe (2430-2460). Con questa è in grado di impostare una partita in modo "intelligente" ma non certo priva di pecche.

Per perfezionarla viene in soccorso la routine numerata da 10000 in poi. Qui si analizzano in particolare le mosse di apertura, che più hanno influenza, nel bene o nel male, sull'esito finale.

Per memorizzare la partita in corso viene adoperata la variabile PA$ che è aggiornata dopo ogni mossa.

Ad esempio se PA$T"15782" significa che ha giocato per primo il giocatore 1 (noi) e ha scelto il 5, poi il giocatore 2 (il computer) ha preso il 7 e così via. Un esempio con inizio del computer:
PA$="2643".

Questa variabile viene poi utilizzata nella routine "strategie livello 2", in modo da evitare nelle mosse d'apertura giocate troppo banali e di sfruttare al meglio eventuali ingenuità dell'avversario.

Con questi accorgimenti il gioco del computer è ora ottimo , anche se ancora non perfetto; infatti avrebbe bisogno di sapere ancora qualche altro "trucco" per giocare in maniera impeccabile.

Sarete in grado di insegnarglieli?

Ovviamente evito di spiegare nei dettagli le linee di gioco seguite dal computer, per non togliere il piacere a quanti vorranno cimentarsi nella sfida.

Dando il RUN verranno mostrate sullo schermo delle carte stilizzate al massimo, ma comunque funzionali per il nostro scopo. Per fare la scelta basta premere il tasto corrispondente al valore indicato dalla carta.

Buon diverimento!

Lo "Sposalizio di Maria Vergine", uno dei capolavori di Raffaello (conservato alla galleria Brera di Milano) è la base da cui sono partiti Daniele Marini e Lorenzo Forges Davanzati per le loro sperimentazioni anamorfiche al computer. Voltare pagine per credere.

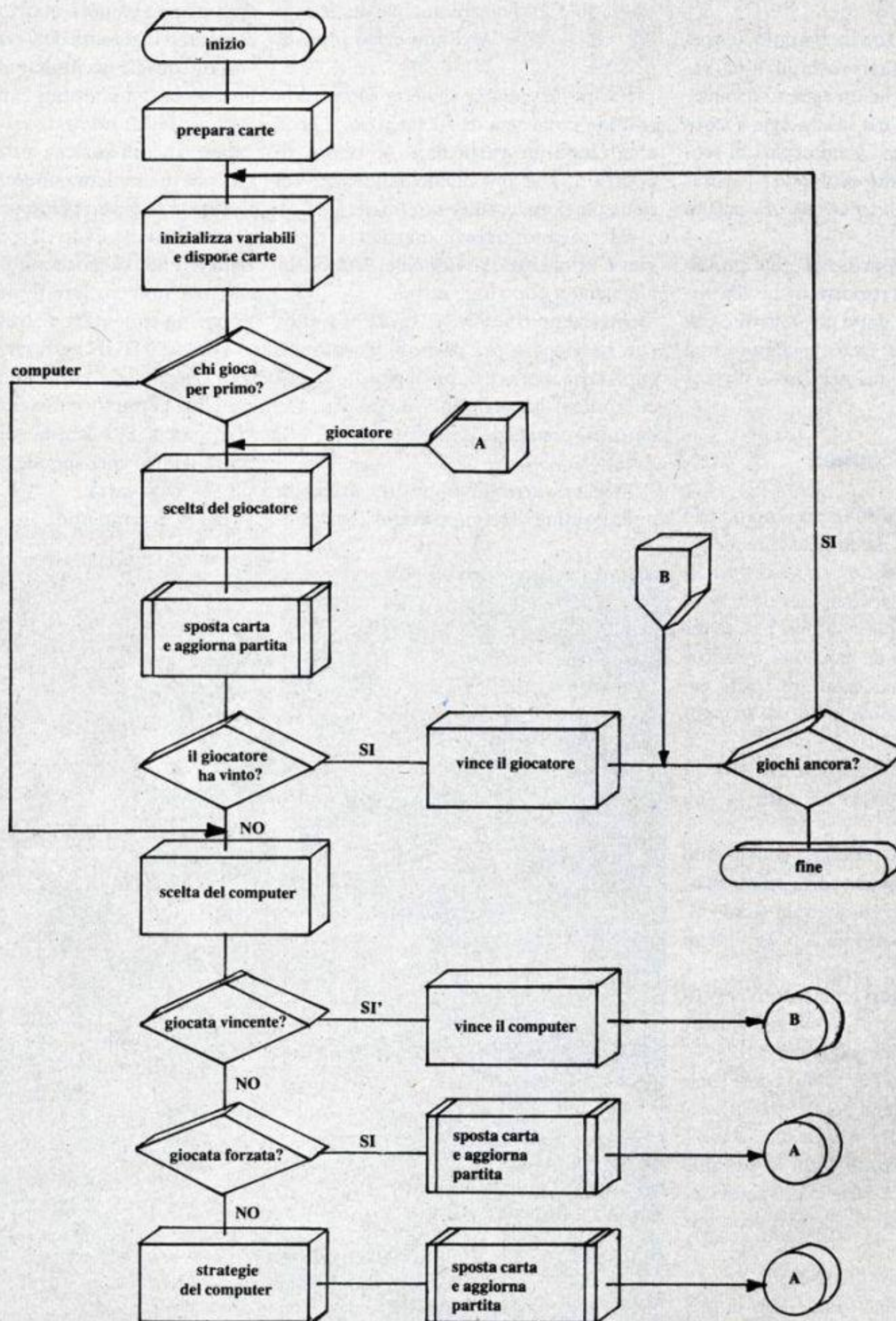



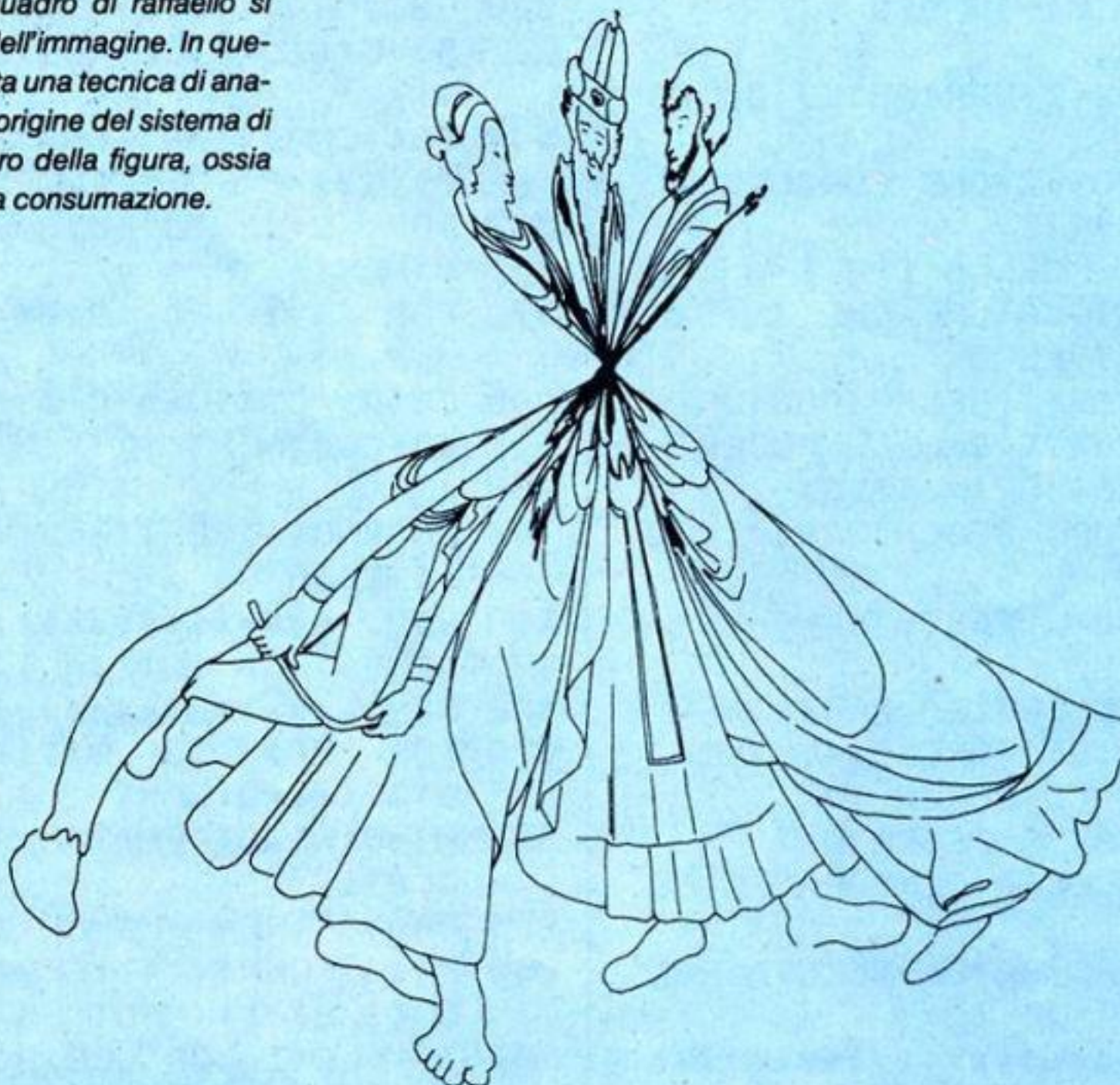
Diagramma a blocchi dell'intero programma. La scelta del computer viene fatta attraverso i blocchi segnati con ✱

Attraverso l'anamorfosi conica, ecco la prima trasformazione dell'originale di Raffaello. I due autori hanno dato all'immagine un titolo significativo: la sottomissione.

```
100 REM  GIOCO DELLE 9 CARTE
110 :
120 REM  BY FLAVIO MOLINARI
125 :
130 REM  C64 - C16
135 :
140 REM  *************************

142 REM  VARIABILI IN USO
144 :
146 REM  Q,K,H,J,Z=VARIABILI DI CO
    MODO
148 REM  D$()=POSIZIONE CURSORE
150 REM  C$()=CARTE
152 REM  C8$ =CANCELLA CARTA
154 REM  PG  =GIOCATORE CHE INIZIA
155 REM  LE  =LIVELLO
156 REM  T1()=CARTE DEL GIOCATORE
158 REM  T2()=CARTE DEL COMPUTER
160 REM  T3()=CARTE IN TAVOLA
162 REM  N1  =NUM. PESCATE GIOCATO
    RE
164 REM  N2  =NUM. PESCATE COMPUTE
    R
166 REM  PA$ =MEMORIZZA PARTITA
168 REM  GC  =GIOCATORE DI TURNO:
    1=GIOC.  2=COMPUTER
170 REM  G   =CARTA PESCATA
172 REM  A   =SCELTA TEMPORANEA CO
    MPUTER
174 REM  U   =VARIABILE SCELTA PSE
    UDOCASUALE
180 REM  *************************
210 DIM  D$(24):D$(1)=CHR$(19)
220 FOR  Q=2  TO  24:D$(Q)=CHR$(19
    )+D$(Q-1)+CHR$(17):NEXT
230 C2$="               ":REM  15
    SPAZI
498 REM  *************************
500 REM       PREPARA CARTE
502 REM  *************************
510 C0$=CHR$(98):C1$=CHR$(17)+CHR$
    (157)+CHR$(157)+CHR$(157)
520 C3$=CHR$(117)+CHR$(96)+CHR$(10
    5)+C1$:C4$=C0$+CHR$(32)+C0$+C1
    $
540 C6$=CHR$(106)+CHR$(96)+CHR$(10
    7):C7$="   ":REM  3 SPAZI
560 FOR  Q=1  TO  5:C8$=C8$+C7$+C1
    $:NEXT
570 FOR  Q=1  TO  9:Q$=MID$(STR$(Q
    ),2,1)
580 C$(Q)=C3$+C4$+C0$+Q$+C0$+C1$+C
    4$+C6$:NEXT
698 :
700 PG=RND(1)*2-1:REM  SCEGLIE CHI
     INIZIA
798 REM  *************************
800 REM        INIZIALIZZA
802 REM  *************************
810 FOR  Q=1  TO  9:T1(Q)=0:T2(Q)=
    0:T3(Q)=Q:NEXT
820 N1=0:N2=0:REM  NUMERO CARTE GI
    OCATE
830 PA$="":REM  MEMORIZZA PARTITA
850 PRINT CHR$(147)SPC(252)"LIVELL
    O (1/2)";:INPUT  LE
860 IF  LE<1  OR  LE>2  THEN  850
998 REM  *************************
1000 REM       DISPONE CARTE
```

```
1002 REM *************************
1010 PRINT CHR$(147)
1020 FOR Q=1 TO 9:PRINT D$(11) T
    AB(Q*3+3)C$(Q):NEXT
1198 :
1200 IF PG>=0 THEN PA$="1":REM
    INIZIA IL GIOCATORE
1210 IF PG<0 THEN PA$="2":GOTO
     1810:REM INIZIA IL COMPUTER
1398 REM *************************
1400 REM    SCELTA DEL GIOCATORE
1402 REM *************************
1410 GC=1:IF N1+N2=9 THEN 5005
1420 PRINT D$( 2) TAB(20)C2$
1430 PRINT D$(22) TAB(20)"SCEGLI ?"
1440 GET Q$:G=VAL(Q$):IF G<1 OR
     G>9 THEN 1440:REM SCELTA
1450 IF T3(G)=0 THEN 1440:REM G
    IA' GIOCATA
1460 GOSUB 6010
1598 REM *************************
1600 REM  IL GIOCATORE HA VINTO ?
1602 REM *************************
1605 SW=0:IF N1<3 THEN 1680
1610 FOR K=1 TO N1-2
1620 FOR H=K+1 TO N1-1
1630 FOR J=H+1 TO N1
1640 IF T1(K)+T1(H)+T1(J)<>15 THE
    N 1660
1650 SW=1:V1=T1(K):V2=T1(H):V3=T1(J
    ):REM COMBINAZIONE VINCENTE
1660 NEXT:NEXT:NEXT
1680 IF SW=1 THEN GOSUB 7010:GO
    TO 5010:REM SI
1798 REM *************************
1800 REM     SCELTA DEL COMPUTER
1802 REM *************************
1810 GC=2:IF N1+N2=9 THEN 5005:R
    EM FINE
1820 PRINT D$(22) TAB(20)C2$
1830 PRINT D$( 2) TAB(20)"STO PENSA
    NDO...":GOSUB 8000
1998 REM *************************
2000 REM     GIOCATA VINCENTE ?
2002 REM *************************
2005 G=0:IF N2<2 THEN 2060
```

I personaggi del quadro di raffaello si uniscono al centro dell'immagine. In questo caso è stata usata una tecnica di anamorfosi sferica con origine del sistema di riferimento nel centro della figura, ossia nell'anello. Il titolo: la consumazione.

```
2010 FOR  K=1  TO  N2-1:FOR  H=K+1
      TO  N2
2020 A=15-T2(K)-T2(H):IF  A>9  OR
     A<0  THEN  A=0
2030 IF  T3(A)=0  THEN  2050:REM  G
     IA' GIOCATA
2040 G=A:V1=G:V2=T2(K):V3=T2(H):REM
      COMB. VINCENTE
2050 NEXT:NEXT
2060 IF  G<>0  THEN  GOSUB  6010:GO
     SUB  7510:GOTO  5010:REM  SI
2198 REM  ************************
2200 REM       GIOCATA FORZATA ?
2202 REM  ************************
2205 IF  N1<2  THEN  2250:REM  NO
2210 FOR  K=1  TO  N1-1:FOR  H=K+1
      TO  N1
2220 A=15-T1(K)-T1(H):IF  A>9  OR
     A<0  THEN  A=0
2230 IF  T3(A)<>0  THEN  G=A
2240 NEXT:NEXT
2250 IF  G<>0  THEN  GOSUB  6010:GO
     TO  1410
2398 REM  ************************
2400 REM   STRATEGIA DEL COMPUTER
2402 REM  ************************
2410 G=0:IF  LE=2  THEN  GOSUB  100
     10:REM  STRATEGIE LIV 2
2420 IF  G<>0  THEN  2490:REM  HA G
     IA' SCELTO
2430 U=2:IF  N1+N2>5  THEN  U=1
2450 G=INT(RND(1)*10/U)*U:REM  SCEL
     TA PSEUDOCASUALE
2460 IF  G=0  THEN  G=5
2470 IF  T3(G)=0  THEN  2450:REM  G
     IA' GIOCATA
2490 GOSUB  6010:GOTO  1410:REM  TO
     RNA AL GIOCATORE
4998 REM  ************************
5000 REM       FINE DELLA PARTITA
5002 REM  ************************
5005 PRINT D$(22) TAB(20)"PARTITA P
     ARI."
5010 GOSUB  8000:PRINT D$(23) TAB(2
     0)"GIOCHIAMO ANCORA ?"
5020 GET  Q$:IF  Q$="S"  THEN  PG=-
     PG:GOTO  810
5030 IF  Q$="N"  THEN  END
5040 GOTO  5020
5998 REM  ************************
6000 REM        SPOSTA CARTA E
6002 REM       AGGIORNA PARTITA
6004 REM  ************************
6010 PRINT D$(11) TAB(G*3+3)C8$:T3(
     G)=0:REM  CANCELLA CARTA
6020 IF  GC=1  THEN  PRINT D$(19) T
     AB(N1*3)C$(G):N1=N1+1:T1(N1)=G
6030 IF  GC=2  THEN  PRINT D$( 2) T
     AB(N2*3)C$(G):N2=N2+1:T2(N2)=G
6040 PA$=PA$+MID$(STR$(G),2,1):RETU
     RN
6998 REM  ************************
7000 REM      VINCE IL GIOCATORE
7002 REM  ************************
7010 FOR  Z=1  TO  N1
7020 IF  T1(Z)=V1  OR  T1(Z)=V2  OR
      T1(Z)=V3  THEN  7040
7030 GOSUB  8000:PRINT D$(19) TAB(Z
     *3-3)C8$:REM  CANCELLA CARTA
7040 NEXT
7060 PRINT D$(22) TAB(20)"BRAVO, HA
     I VINTO !":RETURN
7498 REM  ************************
7500 REM      VINCE IL COMPUTER
7502 REM  ************************
7510 FOR  Z=1  TO  N2
7520 IF  T2(Z)=V1  OR  T2(Z)=V2  OR
      T2(Z)=V3  THEN  7540
7530 GOSUB  8000:PRINT D$( 2) TAB(Z
     *3-3)C8$
7540 NEXT
7560 PRINT D$(22) TAB(20)"HO VINTO
     IO !!":RETURN
7998 :
8000 FOR  Q=1  TO  2000:NEXT:RETURN
     :REM  RITARDO
9998 REM  ***********************
10000 REM   STRATEGIE LIVELLO 2
10002 REM  ***********************
10010 PA=VAL(PA$)
10020 IF  PA=224  OR  PA=226  THEN
      G=8
10030 IF  PA=242  OR  PA=248  THEN
      G=6
10040 IF  PA=262  OR  PA=268  THEN
      G=4
10050 IF  PA=286  OR  PA=284  THEN
      G=2
10200 IF  PA=1258  OR  PA=1852  OR
      PA=1456  OR  PA=1654  THEN  G=
      7
10300 IF  G=0  AND  T3(5)<>0  AND  N
      1+N2>0  THEN  G=5
11000 RETURN
```

DEDALO

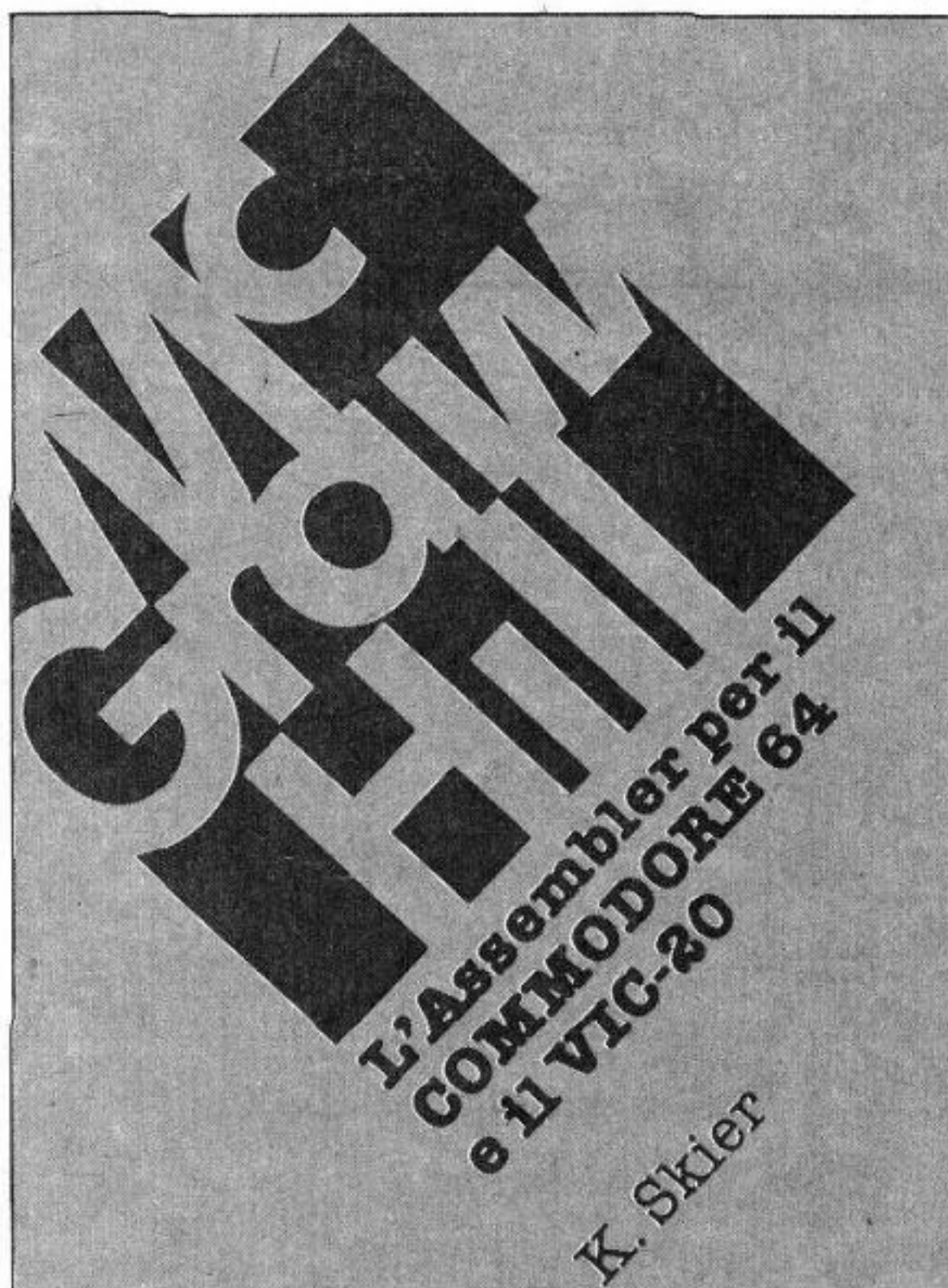
L'obiettivo di questo gioco, destinato agli utenti del computer Vic 20, è riuscire a scappare da un labirinto costituito da molteplici stanze.

Potrete visionare inizialmente la mappa del labirinto e vedere la posizione in cui vi trovate in quel momento.

Se siete dotati di senso dell' orientamento probabilmente riuscirete a trovare la via che vi condurrà al di fuori di questo pericoloso Dedalo.

```
1 POKE 56,28:POKE 55,5:POKE 52,2
  8:POKE 51,5:POKE 649,1
2 DIM M$(2),A$(3):V=7680:W=7174:
  E=506
3 GOSUB 47:TI$="000000":M$(0)="▒
  ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒":M$(1)="▚▚▚▚▚
  ▚▚▚▚▚▚▚▚▚▚▚"
4 M$(2)="▒▒▒▒▒ OTTIMO! ▒▒▒▒▒"
5 A$(0)="ILMJ":A$(1)="JILM":A$(2
  )="MJIL":A$(3)="LMJI":B=0:SP=-
  22:SO=-1:GOTO 20
6 PRINT"[HOME][BIANCO]" TAB(7);T
  I$
7 GET M$: IF M$="" THEN 6
8 IF M$="A" THEN 75
9 FOR I=1 TO 4:IF MID$(A$(B),I,1
  )<>M$ THEN NEXT
10 ON IGOTO 11,12,13,14,15
11 B=0:SP=-22:SO=-1:GOTO 16
12 B=1:SP=1:SO=-22:GOTO 16
13 B=2:SP=22:SO=1:GOTO 16
14 B=3:SP=-1:SO=22:GOTO 16
15 GOTO 7
16 IF M$<>"I" THEN 20
17 IF PEEK(PS+SP)=102 THEN 7
18 IF PS+SP<W+22 THEN 81
19 PS=PS+SP:POKE PS,46
20 PRINT"[CLEAR][BIANCO]"; TAB(7)
   TI$:FOR I=0 TO 20:POKE V+44+I,
   99:NEXT
21 FOR I=0 TO 4
22 IF I>0 AND PEEK(PS+I*SP)=46 TH
   EN POKE V+318-I*44,42
23 CH=0:IF PS+I*SP<W+22 AND PEEK(
   PS+I*SP)=32 THEN CH=1:GOSUB 43
   :GOTO 7
24 IF PEEK(PS+I*SP)=102 THEN GOSU
   B 43:GOTO 7
25 IF PEEK(PS+I*SP+SO)<>102 THEN
   30
26 POKE V+308-I*42,78:POKE V+287-
   I*42,78
27 IF PEEK(PS+I*SP-SO)<>102 THEN
   36
28 POKE V+328-I*46,77:POKE V+305-
   I*46,77
29 NEXT:GOSUB 43:GOTO 7
30 IF I=0 THEN 33
31 C=V+308-I*42
32 POKE C,101:C=C-22:IF C>=V+66 T
   HEN 32
33 POKE C,79:C=V+265-I*42:POKE C-
   1,76:POKE C,122:IF I=0 THEN PO
   KE C-1,100
34 C=C-22:POKE C,103:IF C>V+88 TH
   EN 34
35 POKE C-22,80:GOTO 27
36 IF I=0 THEN 40
37 C=V+328-I*46
38 POKE C,103:C=C-22:IF C>=V+66 T
   HEN 38
39 POKE C,80
```

```
40 C=V+283-I*46:POKE C,76:POKE C+
   1,122:IF I=0 THEN POKE C+I,100
41 C=C-22:POKE C,101:IF C>V+88 TH
   EN 41
42 POKE C-22,79:GOTO 29
43 IF CH=0 THEN KK=INT(RND(1)*5):
   IF KK=1 THEN CH=2
44 K=21-I*4:C=12-I*2:PRINT:FOR I=
   0 TO C
45 IF CH=2 AND K<>17 THEN CH=0
46 PRINT TAB((21-K)/2);LEFT$(M$(C
   H),K):NEXT:RETURN
47 PRINT"[CLEAR][6 DOWN]VUOI LE I
   STRUZIONI?"
48 GET A$:IF A$="" THEN 48
49 IF A$="N" THEN 62
50 POKE 36879,27:PRINTCHR$(14);"[
   BLEU][CLEAR][RVS]*****STAI ATT
   ENTO!****[RVOFF]"
51 PRINT"[DOWN]ORA TI MOSTRO LA P
   IAN-TA DI UN LABIRINTO"
52 PRINT"[DOWN]TU CI SEI DENTRO!"
   :PRINT"[DOWN]LA TUA POSIZIONE
   E'[3 RIGHT][DOWN]INDICATA DALL
   A FRECCIA"
53 PRINT"CHE MOSTRA ANCHE LA":PRI
   NT"[DOWN]TUA DIREZIONE!":PRINT
   "[RVS][2 DOWN]    VEDRAI TUTTO
    IN    "
54 PRINT"[RVS][UP]    3-DIMENSION
   I!      ":PRINT"[DOWN]    PREMI
    UN TASTO"
55 GET R$:IF R$="" THEN 55
56 PRINT"[CLEAR]PER AIUTARTI LASC
   ERAI":PRINT"[DOWN]LE IMPRONTE
   DOVE SEI":PRINT"[DOWN]GIA' PAS
   SATO"
57 PRINT TAB(9)"[DOWN][RVS]¬[RVOF
   F][DOWN][2 LEFT][RVS]└[RVOFF]
   [RVS]L[RVOFF][DOWN][2 LEFT][RV
   S]\[RVOFF]"
58 PRINT"[DOWN]¬=AVANZI DI 1 PASS
   O":PRINT"[DOWN]└=TI GIRI A SIN
   ."
59 PRINT"[DOWN]L=TI GIRI A DESTRA
   ":PRINT"[DOWN]\=TI GIRI INDIET
   RO"
60 PRINT"[DOWN]♠=RINUNCI":PRINT"[
   2 DOWN]     PREMI UN TASTO";
61 GET R$:IF R$="" THEN 61
62 PRINT"[CLEAR]";CHR$(142):POKE
   36879,8:B=INT(RND(0)*20+1):FOR
    I=0 TO 21:IF I=B THEN POKE V+
   I,32:GOTO 64
63 POKE V+I,102
64 NEXT
65 FOR R=1 TO 21:POKE V+R*22,102:
   POKE V+R*22+21,102:FOR C=1 TO
   20:A=INT(RND(1)+.5)
66 IF PEEK(V+R*22+C-22)=102 THEN
   POKE V+R*22+C,102-70*A:GOTO 70
67 IF PEEK(V+R*22+C-21)=32 AND PE
   EK(V+R*22+C-23)=32 THEN POKE V
   +R*22+C,102:GOTO 70
68 IF PEEK(V+R*22+C-23)=32 AND PE
   EK(V+R*22+C-1)=32 THEN POKE V+
   R*22+C,102:GOTO 70
69 POKE V+R*22+C,32
70 NEXT:NEXT:FOR C=0 TO 21:POKE V
   +484+C,102:NEXT:FOR I=0 TO 505
   :POKE W+I,PEEK(V+I):NEXT:C=21-
   B
71 D=C+V+396
72 IF PEEK(D)=102 THEN C=C+1:GOTO
    71
73 PS=W+396+C:POKE PS,46:POKE D,3
   0
74 FOR I=1 TO 30:POKE D,188-PEEK(
   D):FOR M=1 TO 300:NEXT:NEXT:PR
   INT"[CLEAR]":RETURN
75 PRINT"[CLEAR]"
76 FOR I=0 TO 505:POKE V+I,PEEK(W
   +I):NEXT:PRINT"[HOME]":FOR I=0
    TO 20:PRINT"[DOWN]";:NEXT:PRI
   NT"[BIANCO]VUOI GIOCARE?";
77 GET A$:IF A$="" THEN POKE PS+E
   ,220-PEEK(PS+E):GOTO 77
78 IF A$="S" THEN PRINT"[CLEAR]":
   GOTO 3
79 IF A$="N" THEN PRINT"[CLEAR][4
    DOWN][3 RIGHT]CIAO!":POKE 649
   ,10:END
80 GOTO 77
81 A$=TI$:PRINT"[CLEAR][4 DOWN][R
   VS]*****BRAVISSIMO!!*****"
82 PRINT"[3 DOWN]SEI USCITO IN:[D
   OWN]"
83 PRINTLEFT$(A$,2);"ORE ";MID$(A
   $,3,2);"MIN ";RIGHT$(A$,2);"SE
   C"
84 PRINT"[BIANCO][2 DOWN]ORA TI M
   OSTRO LA":PRINT"[BIANCO][DOWN]
   STRADA PERCORSA":FOR I=0 TO 50
   00:NEXT:GOTO 75
```

L'assembler per il C 64 e il Vic 20

Pur se con ritardo, "esce" finalmente il volume sull'Assembler promesso fin dall'anno scorso dalla nota Casa Editrice.

Purtroppo il Vic 20 è nel frattempo uscito di produzione ed il frettoloso lettore potrebbe, a torto, ritenere obsoleto il volume. Riteniamo, al contrario, che il libro (di oltre 360 pagine) può essere un valido aiuto per coloro che desiderano approfondire le proprie conoscenze sul linguaggio macchina e sull'assembler in generale.

Il microprocessore preso in esame è ovviamente il 6502 sia nella principale versione (Vic 20) che nella versione 6510 (Commodore 64).

Il sistema adoperato per avvicinare il lettore ai numerosi argomenti è quello, sperimentato con successo, dei *"piccoli passi"*. *Si inizia* con una trattazione teorica di alcune semplici istruzioni in L.M. e, un pò per volta, si costruisce un gruppo di utility (in linguaggio macchina caricabili da Basic) che costituiranno, alla fine, non solo un utile programma Editor, ma anche un programma modificabile *a volontà dall'utente proprio perchè ne viene* fornita la spiegazione istruzione per istruzione!

Ciascuna subroutine L.M. proposta viene, infatti, attentamente esaminata ed illustrata allo scopo di comprenderne perfettamente il funzionamento.

E' indispensabile che chi utilizza il Vic 20 inserisca l'espansione da almeno 8K di RAM. Chi possiede il C-64, invece, non ha problemi di sorta.

I possessori del C-16 o di altri computer, purtroppo, non potranno apprezzare appieno *le informazioni del volume* dato che i programmi pubblicati sfruttano spesso le numerose routines già presenti nelle ROM.

Il testo vero e proprio è ospitato in 179 pagine. Le rimanenti rappresentano sette appendici tra cui il disassemblato completo del monitor che si può realizzare leggendo il volume.

Nonostante sia di agevole lettura, consigliamo il libro ad utenti non di primo pelo, che abbiano cioè maturato una certa esperienza col linguaggio Basic non perchè tale conoscenza sia indispensabile, ma perchè "parlare" direttamente col cervello di un calcolatore richiede un rigore ed un'abitudine alla logica di programmazione di un certo impegno.

K. Skie: "Assembler per il Commodre 64 e il *Vic 20*". *Edizioni Mc. Graw Hill*
Pagine 365 - L. 35.000

COMMODORE 64 - VIC 20 - PLUS 4 - C 16

FUGA!

di **Luca Galuzzi**

Il difetto dei bellissimi giochi esistenti per il computer è la difficile lettura e comprensione del loro listato.

Nel caso del nostro gioco, invece, la facilità di lettura va a discapito della complessità dello stesso, che si riduce alla presenza sullo schermo di due inseguitori coalizzati per mangiarvi.

Voi, attraverso i tasti I, M, L e J, potete spostarvi sullo schermo rispettivamente in alto, in basso, a destra e a sinistra. Ad ogni vostro "passo" si muovono anche i due inseguitori che però funzionano a singhiozzo, a volte restando fermi ed a volte facendo un "passo" per volta.

Il fatto che gli inseguitori si muovano a singhiozzo vi permette di distanziarli sui rettilinei. Attenzione però: in caso di "curve" hanno la possibilità di muoversi in diagonale, mentre voi non potete farlo e questo chiaramente permette loro di guadagnare terreno nei vostri confronti.

Il gioco poi è reso più interessante dalla presenza di un certo numero di "fiori" sparsi sullo schermo, cogliendo i quali guadagnate punti (purtroppo solo dieci).

Un semplice giochino di inseguimento per prendere confidenza con il Commodore 64 che può essere analizzato a fondo anche da chi si avvicina per le prime volte al computer.

La soluzione per realizzare molti punti consiste nel fare in modo che siano gli inseguitori a mangiare i fiori. Infatti ogni volta che uno di essi passa su un fiore ne compaiono immediatamente altri due, aumentando così la possibilità di incrememtare il punteggio.

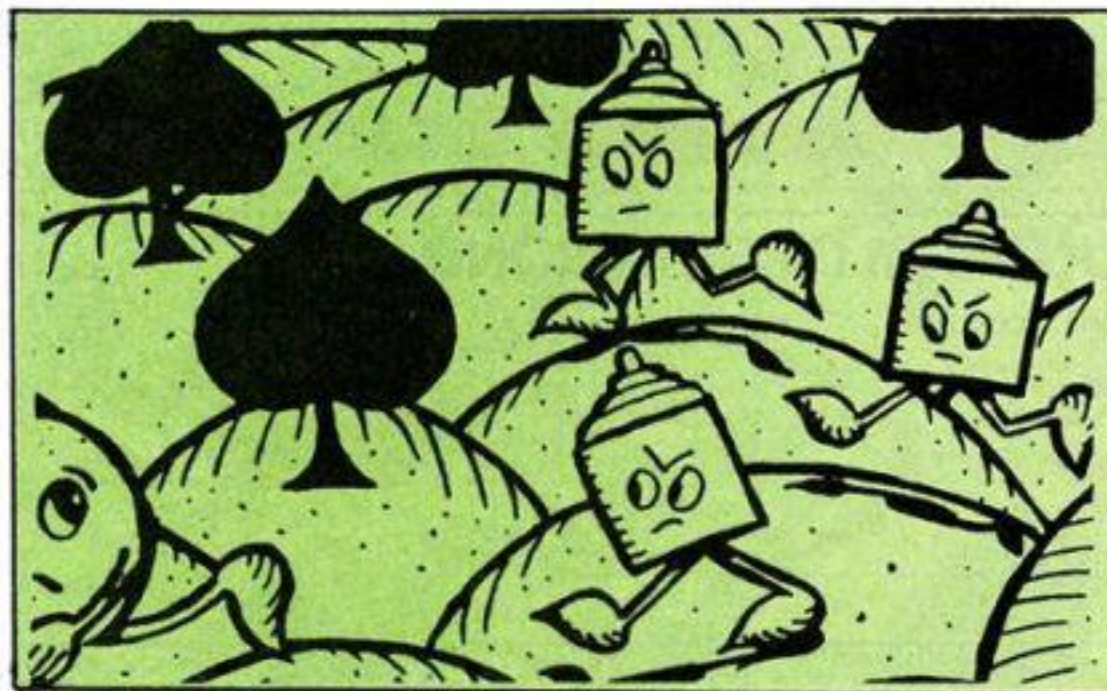
Il funzionamento del programma si basa sul controllo delle coordinate vostre e degli inseguitori. Quando cioè l'ascissa dell'inseguito è maggiore di quella dell'inseguitore, questa viene incrementata di un valore che può essere casualmente uno oppure zero, in modo da dare il movimento a singhiozzo; lo stesso avviene per le altre coordinate.

La stampa dei caratteri grafici avviene attraverso alcune istruzioni di POKE che puntano direttamente ad una locazione di schermo.

Il giochino può essere facilmente modificato ed ampliato da ognuno di voi con l'aggiunta, ad esempio, di ostacoli superabili solo dagli inseguitori, con l'inserimento di qualche "fiore velenoso" o in cento altri modi dipendenti solo dalla vostra fantasia.

```
1000 REM    F U G A !!
1010 REM    GIOCO  PER
1020 REM   COMMODORE 20
1040 REM   BY LUCA GALUZZI
1050 :
1060 CLR :X=12:Y=12:X1=0:Y1=0:X2=22
     :Y2=23
1066 XS=7680:XC=38400
1070 REM  X= N.COLONNA POSIZIONE PE
     DINA
1080 REM  Y= N.RIGA POSIZIONE PEDIN
     A
1090 REM  X1 , X2= N.COLONNA POSIZI
     ONE AVVERSARI
1100 REM  Y1 , Y2= N.RIGA POSIZIONE
      AVVERSARI
1110 PRINTCHR$(147)"TASTI [RVS]-I-J
     -L-M-":PRINT"PREMI UN TASTO"
1120 GET  A$:IF  A$=""  THEN  1120
1130 PRINT CHR$(147)"ATTENDERE": RE
     M  CANCELLA SCHERMO E MESSAGGI
     O
1140 FOR I=XC TO XC+506:POKE I,0:NE
     XT:REM  COLORA CELLE VIDEO IN
     BIANCO
1150 PRINTCHR$(19)" SCAPPA!!"
1160 POKE  650,128:REM  REPEAT AUTO
     MATICO SU TUTTI I TASTI
1170 GOSUB  1710:REM  SALTA A SUBRO
     UTINE DI VISUALIZZAZIONE FIORI
1180 POKE  XS+X+22*Y,81:REM  VISUAL
     IZZA PEDINA IN COORD. X,Y
1190 POKE  XS+X2+22*Y2,87:POKE  XS+
     X1+22*Y1,87:REM  VISUALIZZA NE
     MICO
1200 REM  ATTESA PRESSIONE DI UN TA
     STO
1210 GET  A$:IF  A$=""  THEN  1210
1220 REM  CALCOLO SPOSTAMENTO CASUA
     LE INSEGUITORI
1230 A=INT(RND(1)*2)
1240 POKE  XS+X+22*Y,32:REM  PRINT
     AT X,Y
1250 REM  CONTROLLO TASTI DI MOVIME
     NTO
1260 IF  A$="I"  AND  Y>0  THEN  Y=
     Y-1 :REM  IN ALTO
1270 IF  A$="M"  AND  Y<22  THEN  Y
     =Y+1:REM  IN BASSO
1280 IF  A$="J"  AND  X>0  THEN  X=
     X-1 :REM  SINISTRA
1290 IF  A$="L"  AND  X<21  THEN  X
     =X+1:REM  DESTRA
1300 REM  CONTROLLO "CATTURA" FIORI
1310 IF PEEK(XS+X+22*Y)=88 THEN P=P
     +50:POKE (XS+X+22*Y),32
1320 POKE XS+X+22*Y,81
1330 POKE XS+X1+22*Y1,32
1340 POKE XS+X2+22*Y2,32
1350 REM  SPOSTAMENTO INSEGUITORI
1360 IF X1<X THEN X1=X1+A:GOTO 1380
1370 X1=X1-A
1380 IF Y1<Y THEN Y1=Y1+A:GOTO 1400
1390 Y1=Y1-A
1400 IF X2<X THEN X2=X2+A:GOTO 1420
1410 X2=X2-A
1420 IF Y2<Y THEN Y2=Y2+A:GOTO 1500
1430 Y2=Y2-A
1440 IF PEEK(XS+X1+22*Y1)=88 THEN G
     OSUB 1660
1450 IF PEEK(XS+X1+1)=88 THEN GOSUB
      1660
1460 IF PEEK(XS+X1-1)=88 THEN GOSUB
      1660
1470 IF PEEK(XS+X1-22*Y1)=88 THEN G
     OSUB 1660
1480 IF PEEK(XS+X2-22*Y2)=88 THEN G
     OSUB 1660
1490 IF PEEK(XS+X2+22*Y2)=88 THEN G
     OSUB 1660
1500 POKE  XS+X1+22*Y1,87
1510 POKE  XS+X2+22*Y2,87
1520 REM  CONTROLLO SE INSEGUITO E'
      RAGGIUNTO
1530 IF  (X1=X  AND  Y1=Y)  OR  (X2
     =X  AND  Y2=Y)  THEN  1590
1540 PRINT CHR$(19)" PUNTEGGIO : ";
     P
1550 GOTO  1200:REM  FINE CICLO
1560 :
1570 REM  *** GNAMM !!!
1580 :
1590 POKE  XS+X+22*Y,86
1600 FOR  D=1  TO  1000:NEXT
1610 POKE  198,0:REM  SVUOTA IL BUF
     FER DI CASSETTA
1620 GET  A$:IF  A$=""  THEN  1620
1630 GOTO  1060
1640 :
1650 REM  *** RIGENERAZIONE FIORI
1660 POKE  (INT(RND(1)*506)+XS),88
1670 POKE  (INT(RND(1)*506)+XS),88
1680 RETURN
1690 :
1700 REM  *** CREAZIONE  CAMPO
1710 FOR  C=1  TO  20:REM  20=NUMER
     O INIZIALE DI FIORI
1720 POKE  (INT(RND(1)*506)+XS),88:
     REM  STAMPA FIORE SU VIDEO
1730 NEXT:RETURN
```

```
1000 REM   F U G A !!
1010 REM   GIOCO  PER
1020 REM   COMMODORE 64
1030 REM   C-16 & PLUS 4
1040 REM   BY LUCA GALUZZI
1050 :
1060 CLR :X=30:Y=12:X1=0:Y1=0:X2=39
     :Y2=23
1062 PRINT"[CLEAR]1: COMMODORE 64"
1063 PRINT"2: C-16  PLUS 4"
1064 GET A$:IF A$="" THEN 1064
1065 IF A$="1" THEN XS=1024:XC=5529
     6
1066 IF A$="2" THEN XS=3072:XC=2048
1070 REM  X= N.COLONNA POSIZIONE PE
     DINA
1080 REM  Y= N.RIGA POSIZIONE PEDIN
     A
1090 REM  X1 , X2= N.COLONNA POSIZI
     ONE AVVERSARI
1100 REM  Y1 , Y2= N.RIGA POSIZIONE
      AVVERSARI
1110 PRINTCHR$(147)"TASTI [RVS]-I-J
     -L-M-":PRINT"PREMI UN TASTO"
1120 GET  A$:IF  A$=""  THEN  1120
1130 PRINT CHR$(147)"ATTENDERE": RE
     M  CANCELLA SCHERMO E MESSAGGI
     O
1140 FOR I=XC TO XC+1000:POKE I,1:N
     EXT:REM  COLORA CELLE VIDEO IN
      BIANCO
1150 PRINTCHR$(19)" SCAPPA!!"
1160 POKE  650,128:REM  REPEAT AUTO
     MATICO SU TUTTI I TASTI
1170 GOSUB  1710:REM  SALTA A SUBRO
     UTINE DI VISUALIZZAZIONE FIORI
1180 POKE  XS+X+40*Y,81:REM  VISUAL
     IZZA PEDINA IN COORD. X,Y
1190 POKE  XS+X2+40*Y2,87:POKE  XS+
     X1+40*Y1,87:REM  VISUALIZZA NE
     MICO
1200 REM  ATTESA PRESSIONE DI UN TA
     STO
1210 GET  A$:IF  A$=""  THEN  1210
1220 REM  CALCOLO SPOSTAMENTO CASUA
     LE INSEGUITORI
1230 A=INT(RND(1)*2)
1240 POKE  XS+X+40*Y,32:REM  PRINT
     AT X,Y
1250 REM  CONTROLLO TASTI DI MOVIME
     NTO
1260 IF  A$="I"  AND  Y>0  THEN  Y=
     Y-1 :REM  IN ALTO
1270 IF  A$="M"  AND  Y<24  THEN  Y
     =Y+1:REM  IN BASSO
1280 IF  A$="J"  AND  X>0  THEN  X=
     X-1 :REM  SINISTRA
1290 IF  A$="L"  AND  X<39  THEN  X
     =X+1:REM  DESTRA
1300 REM  CONTROLLO "CATTURA" FIORI
1310 IF PEEK(XS+X+40*Y)=88 THEN P=P
     +50:POKE (XS+X+40*Y),32
1320 POKE XS+X+40*Y,81
1330 POKE XS+X1+40*Y1,32
1340 POKE XS+X2+40*Y2,32
1350 REM  SPOSTAMENTO INSEGUITORI
1360 IF X1<X THEN X1=X1+A:GOTO 1380
1370 X1=X1-A
1380 IF Y1<Y THEN Y1=Y1+A:GOTO 1400
1390 Y1=Y1-A
1400 IF X2<X THEN X2=X2+A:GOTO 1420
1410 X2=X2-A
1420 IF Y2<Y THEN Y2=Y2+A:GOTO 1500
1430 Y2=Y2-A
1440 IF PEEK(XS+X1+40*Y1)=88 THEN G
     OSUB 1660
1450 IF PEEK(XS+X1+1)=88 THEN GOSUB
      1660
1460 IF PEEK(XS+X1-1)=88 THEN GOSUB
      1660
1470 IF PEEK(XS+X1-40*Y1)=88 THEN G
     OSUB 1660
1480 IF PEEK(XS+X2-40*Y2)=88 THEN G
     OSUB 1660
1490 IF PEEK(XS+X2+40*Y2)=88 THEN G
     OSUB 1660
1500 POKE  XS+X1+40*Y1,87
1510 POKE  XS+X2+40*Y2,87
1520 REM  CONTROLLO SE INSEGUITO E'
      RAGGIUNTO
1530 IF  (X1=X  AND  Y1=Y)  OR  (X2
     =X  AND  Y2=Y)  THEN  1590
1540 PRINT CHR$(19)" PUNTEGGIO : ";
     P
1550 GOTO  1200:REM  FINE CICLO
1560 :
1570 REM  *** GNAMM !!!
1580 :
1590 POKE  XS+X+40*Y,86
1600 FOR  D=1  TO  1000:NEXT
1610 POKE  198,0:REM  SVUOTA IL BUF
     FER DI CASSETTA
1620 GET  A$:IF  A$=""  THEN  1620
1630 GOTO  1060
1640 :
1650 REM  *** RIGENERAZIONE FIORI
1660 POKE  (INT(RND(1)*1000)+XS),88
1670 POKE  (INT(RND(1)*1000)+XS),88
1680 RETURN
1690 :
1700 REM  *** CREAZIONE  CAMPO
1710 FOR  C=1  TO  20:REM  20=NUMER
     O INIZIALE DI FIORI
1720 POKE  (INT(RND(1)*1000)+XS),88
     :REM  STAMPA FIORE SU VIDEO
1730 NEXT:RETURN
```

SISTEMA OPERATIVO

IL KERNAL

di **Giovanni Verrelli**

Un'applicazione Turbo Tape a caricamento rapido.

Prima di concludere il discorso sul Kernal esaminando le ultime routine che potrebbero avere una certa importanza pratica ai fini della programmazione, soffermiamoci un attimo su quanto detto in precedenza e più precisamente sulle routine SAVE e accessorie.

E' nostra intenzione proporre un esempio pratico sul suo corretto uso, fornendo allo stesso tempo un ausilio che sicuramente sarà da molti ben accetto.

Come sicuramente ricorderete, la routine SAVE è in grado di salvare su dispositivo di memorizzazione esterno qualsiasi zona di memoria centrale, generando in questa fase un file programma richiamabile dal semplice comando LOAD del Basic.

Cercheremo allora di rendere ancor più efficiente un programma che rappresenti, per tutti i possessori di Commodore 64 e registratore, un autentico gioiello: il Turbo Tape, che la nostra rivista su cassetta **Commodore Club** ha più volte inserito nei suoi nastri.

Lo scopo, per i pochi "sessantaquattristi" che ancora non lo conoscessero, è minimizzare i tempi di attesa durante il caricamento di programmi da registratore, aumentando notevolmente (fino a 10 volte!) la velocità di trasferimento dati tra la periferica e l'unità centrale.

Tale programma però, oltre ad alcune

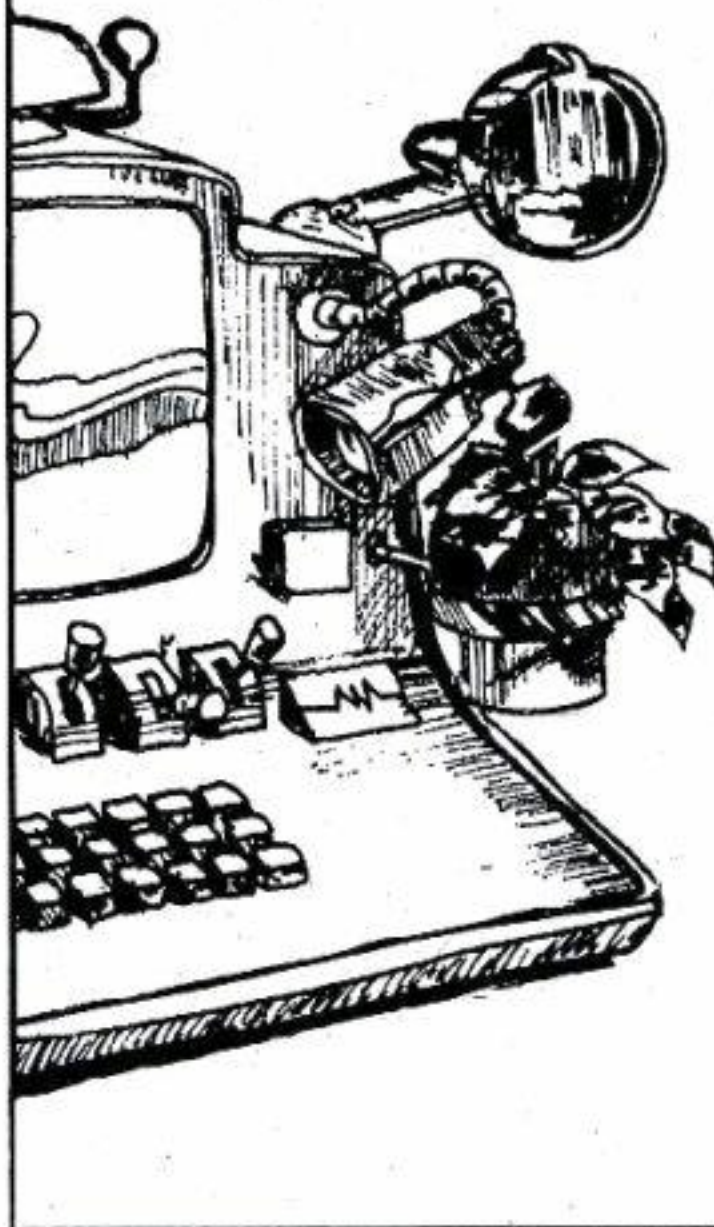
routine che permettono questa "accelerazione", contiene una schermata iniziale di istruzioni che, a lungo andare, potrebbe provocare (e quasi sempre provoca) un certo fastidio e il desiderio di eliminarla. Facendo ciò, inoltre, si diminuirebbe anche il tempo del caricamento iniziale, direttamente proporzionale alla lunghezza del programma.

Dobbiamo quindi salvare su memoria di massa una nuova versione del Turbo Tape, identica alla precedente tranne che per le istruzioni iniziali, impiegando meno tempo per la memorizzazione e, conseguentemente, per la successiva rilettura.

Prima di iniziare però bisogna sapere qual è la zona di memoria (contenente le routine del Turbo) da salvare. Questa è stata individuata nelle locazioni da 50000 ($C350 in esadecimale) a 50774 ($C656) compreso.

Sarà fornita solamente la versione in BASIC, accessibile a tutti: quella in assembly può essere un utile esercizio personale per i pochi (o molti?) conoscitori del linguaggio macchina.

Come procedere

In primo luogo utilizzeremo la routine SETLFS per impostare il numero logico del file, il numero della periferica e l'indirizzo secondario, effettuando:

```
10 POKE780,10 NUMERO
   LOGICO DEL FILE=10
20 POKE781,1:REM POKE
   781,8 PER DISCO
30 POKE782,1:REM
   INDIRIZZO SECON-
   DARIO=1
40 SYS 65466
```

Fatto questo, utilizzeremo SETNAM per predisporre il nome del file e, stabilito di usare la stringa "NEW TURBO TAPE" come nome del programma, effettueremo:

```
50 NF$=NF$+"NEW TURBO
   TAPE":REM NF$=NOME
   DEL PROGRAMMA
60 LI=PEEK(53)+PEEK
   (54)*256
70   LI=LI-LEN(NF$):
   REM LI=LOCAZIONE
   INIZIALE AREA DI
   MEMORIZZAZIONE DI
   NF$
80 POKE782,LI/256:REM
   BYTE ALTO LI
90 POKE781,LI-PEEK
   (782)*256:REM BYTE
   BASSO LI
100 POKE780,LEN(NF$):
    REM LUNGHEZZA NO-
    ME DEL PROGRAMMA
110 SYS65469
```

A questo punto non ci resta che concludere con la routine SAVE:

```
120 IN=50000:REM IN=
    INDIRIZZO AREA DA
    SALVARE
130 POKE254,IN/256:
    REM BYTE ALTO IN
140 POKE253,IN-PEEK
    (254)*256:REM
    BYTE IN
150 POKE780,253:REM
    253=LOCAZIONE CON
    TENENTE IL BYTE
    BASSO DI IN
160 FI=50774+1:REM
    FI=FINE AREA DA
    SALVARE + 1
170 POKE782,FI/256:
    REM BYTE ALTO FI
180 POKE781,FI-PEEK
    (782)*256:REM BY-
    TE BASSO FI
190 SYS65496
200 END
```

Per l'uso del programma, da copiare interamente dalla linea 10 alla 200 (si possono omettere solo le REM), basta caricare in memoria il "Turbo Tape", digitare il listato precedente, attivarlo col RUN e premere i tasti RECORD & PLAY quando richiesto (logicamente deve essere stata inserita precedentemente l'idonea cassetta vergine).

Col disk drive non ci sarà alcuna richiesta di pressione di tasti, ma si vedrà solo lo stesso lavorare (anche in questo caso deve già essere presente l'idoneo dischetto).

Dopo queste operazioni avremo su cassetta (o dischetto) la nuova versione del Turbo Tape utilizzabile con la sequenza di istruzioni:

1/ LOAD"NEW TURBO TAPE" (LOAD"NEW TURBO TAPE",8,1 per il disco) e RETURN
2/ SYS 50000 e RETURN
3/ NEW e RETURN

Chiusa la parentesi sul SAVE, possiamo ora esaminare le ultime routine del Kernal con le quali concluderemo la nostra serie di articoli sul sistema operativo del C64 (e Vic 20).

Queste routine conclusive rendono possibile l'effettuazione in linguaggio macchina di alcuni comandi molto comuni in Basic: per questo motivo saranno fornite solamente le istruzioni assembly per un loro corretto impiego.

La routine "OPEN"

Cominciamo con la descrizione della routine OPEN, il cui indirizzo di chiamata è locato in 65472 ($FFC0 in esadecimale). E' usata per aprire un file logico che, una volta predisposto, può essere usato per le operazioni di I/O.

Anch'essa, come LOAD e SAVE, ha bisogno delle routine preparatorie SETLFS e SETNAM per impostare, rispettivamente, il file logico e il nome del file a cui si vuole accedere.

Perciò, volendo effettuare in linguaggio macchina l'equivalente dell'istruzione Basic:

OPEN15,8,15,"......"

basta usare la routine SETLFS per impostare il numero del file logico (2 nel nostro caso), l'indirizzo del dispositivo (8) e l'indirizzo secondario (15), la routine SETNAM per predisporre il nome del file (che nel nostro caso è una serie di puntini da sostituire appropriatamente) ed infine chiamare la routine con:

JSR $FFC0.

Per chiudere il file aperto con la routine OPEN si può far uso della routine CLOSE, il cui indirizzo di chiamata si trova in 65475 ($FFC3). Per il suo uso basta caricare l'accumulatore con il numero del file logico da chiudere (lo stesso usato nella OPEN) e saltare all'indirizzo di chiamata con JSR.

Perciò, per chiudere il file aperto nell'esempio precedente, cioè per effettuare l'istruzione Basic CLOSE 2, bisogna eseguire:

LDA #$02
JSR $FFC3.

Dopo aver aperto un file con una periferica, bisogna specificare se esso sarà usato per operazioni di input oppure per operazioni di output mediante l'impiego delle routine CHKIN e CHKOUT.

La prima, il cui indirizzo di chiamata si trova in 65478 ($FFC6), è usata per stabilire che il file aperto nella OPEN sarà utilizzato per trasferire dati da periferica a unità centrale (input); con la seconda (CHKOUT), il cui indirizzo di chiamata è locato in 65481 ($FFC9), si stabilisce che il file sarà utilizzato per il trasferimento dati da unità centrale a periferica (output).

Logicamente, affinchè queste routine possano produrre l'effetto desiderato, è necessario che il dispositivo aperto sul canale sia effettivamente di input per CHKIN e di output per CHKOUT, altrimenti si verifica un errore con la conseguente perdita di tutta la procedura.

Solo con i dispositivi di I/O standard

(tastiera e video) la chiamata a queste routine non è necessaria e per lo scambio dei dati bisogna solo chiamare le routine a ciò abilitate.

L'uso di CHKIN e CHKOUT è molto semplice: basta caricare, dopo aver chiamato la routine OPEN, il registro indice X con il numero del file logico usato nell'istruzione di apertura, quindi saltare con il comando JSR all'indirizzo di chiamata.

Perciò, volendo predisporre il file aperto nel primo esempio come canale di input, basta effettuare:

```
LDX+ $02
JSR $FFC6
```

Per predisporlo come canale di output bisogna effettuare:

```
LDX+ $02
JSR $FFC9.
```

Esaminiamo ora la routine CHROUT, il cui indirizzo di chiamata è situato in 65490 ($FFD2). Suo Scopo è inviare in output un carattere su un canale già aperto dalle routine OPEN e CHKOUT.

Se la chiamata a queste ultime due routine non viene effettuata, il dato viene inviato allo schermo. Affinchè l'invio possa essere effettuato, bisogna caricare l'accumulatore col byte del dato e poi saltare all'indirizzo di chiamata sempre con il comando JSR.

Risulta chiaro che in condizioni normali, quando cioè non è stato aperto altro canale di output, l'impiego di questa routine sortisce lo stesso effetto del comando BASIC PRINT, in quanto capace di stampare sullo schermo qualsiasi dato il cui codice ASCII sia stato inserito nell'accumulatore prima del salto di chiamata.

Simulazione L.M. del comando PRINT

Volendo stampare su schermo cancellato la stringa "ESEMPIO", bisogna per prima cosa trascrivere in un'opportuna zona di memoria il codice ASCII del comando che effettua la pulizia dello schermo unitamente a quello di ogni singolo carattere che forma la parola "ESEMPIO", seguito da un piccolo ciclo con cui si carica l'accumulatore con i vari dati da visualizzare. Quindi si salta all'indirizzo di chiamata di CHROUT.

In pratica, scegliendo di cominciare dalla locazione 49152 ($C000), avremo:

```
:C000 93 45 53 45 4D 50 49 4F
:C008 00
C009 LDY +$00
C00B LDA $C000,Y
C00E BEQ $C017
C010 JSR $FFd2
C013 INY
C014 JMP $C00B
C017 RTS
```

Questo stesso programma, con l'aggiunta delle idonee istruzioni iniziali con le quali si apre un canale di output con la stampante, può essere utilizzato anche per effettuare delle stampe su carta.

Sempre con l'impiego delle routine finora viste è possibile effettuare in linguaggio macchina tutta quella serie di comandi con i quali si ordina al disk drive di compiere determinate operazioni (ad esempio la VALIDATE) sul dischetto.

Infatti, tenendo presente che questi comandi sono effettuati mediante l'invio all'unità a dischi di opportune stringhe, basterà aprire un canale di output con la periferica in oggetto, inviare la stringa corrispondente all'iniziale del comando che si vuole far eseguire seguita dagli eventuali altri parametri, infine riportare il tutto alle condizioni iniziali e terminare.

In pratica, per effettuare la VALIDATE da linguaggio macchina, bisogna effettuare:

```
C000 LDA #$02
C002 LDX #$08
C004 LDY #$0F
C006 JSR $FFBA
C009 LDA #$00
C00B JSR $FFBD
C00E JSR $FFC0
C011 LDX #$02
C013 JSR $FFC9
C016 LDA #$56
C018 JSR $FFD2
C01B JSR $FFE7
C01E RTS
```

Commentiamo brevemente il listato.

In primo luogo si apre un canale col disco (in Basic avremmo avuto: OPEN2,8,15) usando la routine SETLFS(C000-C008) per impostare numero del file logico (2), dispositivo (8) ed indirizzo secondario (15), la routine SETNAM (C009-C00D) per specificare il nome del file (che in pratica non c'è e perciò abbiamo l'istruzione LDA + $00) e la routine OPEN (C00E-C010) per aprire effettivamente tale file.

Fatto questo, si seleziona il file appena aperto come canale di output (C011-C015) e si trasmette una "V" (codice ASCII $56, 96 in decimale) su tale canale (C016-C01A), seguito dal salto alla routine CLALL che chiude tutti i file aperti e seleziona i canali di I/O di default e dall'RTS finale (da eliminare se il programma continua).

Per gli altri comandi del disco bisogna sostituire l'istruzione della linea C016 con un ciclo col quale si invia, carattere per carattere, la stringa contenente il comando da impartire (per la formattazione, ad esempio, tale stringa potrebbe essere "N0:PIPPO,00").

Continuiamo con la routine CHRIN, avente indirizzo di chiamata in 65487 ($FFCF), per mezzo della quale si preleva un byte del dato da un canale aperto dalla routine OPEN e predisposto come canale di input dalla routine CHKIN.

Anche in questo caso, se la chiamata a queste due routine non viene effettuata, si prevede che i dati provengano dalla tastiera.

Dopo la chiamata a CHRIN, il byte del dato viene sistemato nell'accumulatore.

Se l'input proviene da tastiera, fino a quando non è premuto il tasto RETURN, viene attivato sullo schermo il cursore con possibilità di immettere fino a 80 caratteri.

Simulazione L.M. di INPUT

Volendo simulare in linguaggio macchina l'istruzione Basic Input, dopo aver stabilito in quale zona di memoria conservare i dati da elaborare successiva-

mente (nell'esempio si è scelto di cominciare dalla locazione $C050), basta il seguente semplice programmino:

```
C000 LDX #$00
C002 JSR $FFCF
C005 STA $C050,X
C008 INX
C009 CMP #$0D
C00B BNE #C002
C00D RTS
```

Con esso si preleva un carattere da tastiera e lo si memorizza nell'area scelta precedentemente fino a quando non si preme il tasto RETURN (avente come codice ASCII il valore 13, $0D in esadecimale), nel qual caso si termina con l'RTS finale.

Qualora in input siano assegnati più di 80 caratteri, saranno prelevati solamente quelli digitati dalla posizione 81 in poi con conseguente perdita dei primi 80 iniziali e risultati non certo soddisfacenti.

La variabile ST

Prima di evidenziare un'altra tipica applicazione di CHRIN esaminiamo la routine READST, indirizzo di chiamata 65463 ($FFB7), usata per leggere la "parola di stato".

Per "parola di stato" si intende una variabile, usata in tutti i computer Commodore ed identificata dalle lettere ST, usata dal sistema (e per questo motivo "riservata") per restituire informazioni sull'esito dell'ultima operazione di I/O eseguita.

Questa routine, perciò, restituisce informazioni sullo stato del dispositivo e sugli errori che si sono verificati durante le operazioni di I/O. Tali "notizie" sono riportate nell'accumulatore che, proprio per questo motivo, può assumere diversi valori, evidenziati dalla tabella di figura 1.

I più importanti sono lo 0, che si incontra quando l'operazione è andata bene e il 64, indicante che è stato letto l'ultimo carattere di un file e che perciò i dati da leggere sono terminati.

Per l'uso basta chiamare la routine con JSR e decodificare l'informazione contenuta nell'accumulatore agendo in conseguenza del valore ivi presente.

Ora possiamo vedere come è possibile usare la routine CHRIN per conoscere il tipo di errore verificatosi quando la spia rossa del disk drive comincia a lampeggiare.

Come noto, in Basic ciò è possibile mediante il programma:

```
10 OPEN2,8,15:INPUT+2,A,B$,C,D
20 PRINT A,B$,C,D:CLOSE2
```

che, una volta attivato col RUN, restituisce sullo schermo codice, messaggio d'errore ed eventuale traccia e settore dove si è verificato.

Per effettuare lo stesso lavoro in linguaggio macchina, abbiamo bisogno del seguente programma:

Con esso in primo luogo si stabilisce un canale di comunicazione col drive (esattamente come avevamo fatto con l'esempio della VALIDATE), usando le routine SETLFS, SETNAM e OPEN (C000-C010), quindi si seleziona il canale appena aperto come canale di input (C011-C015).

A questo punto, tramite un ciclo (C016-C022), si legge un carattere dal disco, lo si stampa sullo schermo e si controlla, mediante il salto alla routine READST, se siamo arrivati alla lettura dell'ultimo carattere del messaggio confrontando il valore $40 (64 in decimale) con quello presente nell'accumulatore. Se questi due valori sono differenti, si torna alla lettura e scrittura di un altro carattere e al successivo controllo di fine file. Quando i due valori sono uguali, si salta alla routine che ristabilisce i canali di I/O di default (presenti al momento dell'accensione) e chiude tutti i file aperti. Si termina con l'RTS finale.

Con questo programma si conclude il nostro viaggio all'interno del Kernal. Le rimanenti routine (in totale sono 39), possono essere facilmente studiate personalmente mediante la semplice lettura della Guida di Riferimento del Programmatore, pagina 5.47 e seguenti.

```
C000 LDA #$02
C002 LDX #$08
C004 LDY #$0F
C006 JSR $FFBA
C009 LDA #$00
C00B JSR $FFBD
C00E JSR $FFC0
C011 LDX #$02
C013 JSR $FFC6
C016 JSR $FFCF
C019 JSR $FFD2
C01C JSR $FFB7
C01F CMP #$40
C021 BNE $C016
C023 JSR $FFE7
C026 RTS
```

Figura 1: tabella degli errori riscontrabili con l'uso di READST.

POS. BIT	VALORE DECIM.	LETTURA DA REGISTRATORE	BUS SERIALE I/O	VERIFY+LOAD NASTRO
0	1		Supero Tempo scrittura	
1	2		Supero Tempo lettura	
2	4	Blocco corto		Blocco corto
3	8	Blocco lungo		Blocco lungo
4	16	Errore lettura irrecuperabile		Qualsiasi errore
5	32	Errore checksum		Err.checksum
6	64	Fine file	Fine linea	
7	-128	Fine nastro	Dispositivo non presente	Fine nastro

Figura 2: routine del Kernal richiamabili dall'utente (con l'asterisco sono indicate quelle esaminate durante la nostra serie di articoli)

INDIRIZZO HEX	INDIRIZZO DEC	NOME	DESCRIZIONE
$FFB1	65409	CINT	Inizializza l'editor di schermo
$FF84	65412	IOINIT	Inizializza i dispositivi di I/O
$FF87	65415	RAMTAS	Esegue un test sulla RAM
$FF8A	65418	RESTOR	Riprist. i vettori di default di I/O
$FF8D	65421	VECTOR	Legge/Imposta i vettori di I/O
$FF90	65424	SETMSG	* Controlla i messaggi del Kernal
$FF93	65427	SECOND	Invia l'indir. secondario per LISTEN
$FF96	65460	TKSA	Invia l'indirizzo secondario per TALK
$FF99	65433	MEMTOP	Legge/Imposta la cima della RAM
$FF9C	65436	MEMBOT	Legge/Imposta la base della RAM
$FF9F	65439	SCNKEY	Esegue la scansione della tastiera
$FFA2	65442	SETTMO	Imposta il supero tempo del bus IEEE
$FFA5	65445	ACPTR	Preleva un byte dal bus seriale
$FFA8	65448	CIOUT	Trasmette un byte sul bus seriale
$FFAB	65451	UNTLK	Invia un comando di non-colloquio
$FFAE	65454	UNLSN	Invia un comando di non-ascolto
$FFB1	65457	LISTEN	Ordina di prepararsi a ricevere dati
$FFB4	65460	TALK	Ordina di prepararsi a trasmett. dati
$FFB7	65463	READST	* Legge la parola di stato di I/O
$FFBA	65466	SETLFS	* Predispone un file logico
$FFBD	65469	SETNAM	* Imposta il nome del file
$FFC0	65472	OPEN	* Apre un file logico
$FFC3	65475	CLOSE	* Chiude un file logico specifico
$FFC6	65478	CHKIN	* Apre un canale di input
$FFC9	65481	CHKOUT	* Apre un canale di output
$FFCC	65484	CLRCHN	Chiude i canali di input e di output
$FFCF	65487	CHRIN	* Preleva un byte dal canale di input
$FFD2	65490	CHROUT	* Invia un byte sul canale di output
$FFD5	65493	LOAD	* Carica/Verifica la RAM da periferica
$FFD8	65496	SAVE	* Salva la RAM su un dispositivo
$FFDB	65499	SETTIM	Imposta il clock di sistema
$FFDE	65502	RDTIM	Legge il clock di sistema
$FFE1	65505	STOP	Controlla la pressione del tasto STOP
$FFE4	65508	GETIN	Preleva un carattere
$FFE7	65511	CLALL	* Chiude tutti i canali ed i file
$FFEA	65514	UDTIM	Aggiorna il clock di sistema
$FFED	65517	SCREEN	* Riporta il formato dello schermo
$FFF0	65520	PLOT	* Legge/Imposta la posiz. del cursore
$FFF3	65523	IOBASE	Definisce la zona di memoria per I/O

VIDEO ERGO CHIP

Considerazioni e note sulla periferica più familiare.

Per qualsiasi sistema, il video rappresenta una parte insostituibile: è infatti tramite lui che possiamo comunicare col computer.

La moderna tecnologia mette oggi a disposizione vari tipi di schermi, ma il primato spetta sempre al vecchio "tubo a raggi catodici", quello usato, per intenderci, nei normali televisori.

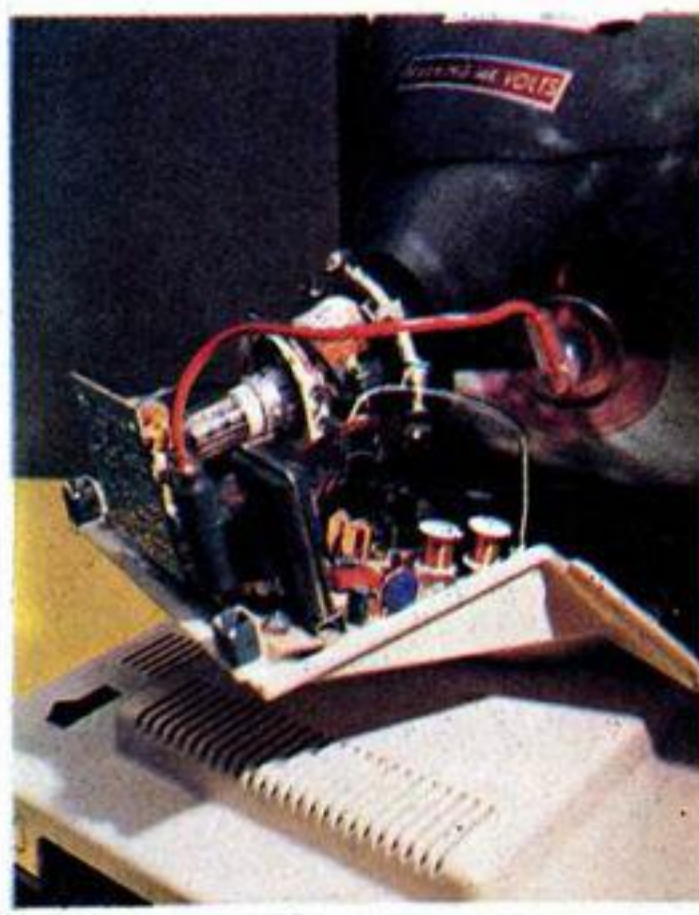
Vediamo ora, molto sinteticamente, il funzionamento di questi tubi.

Un po' di teoria

La loro forma è molto simile a quella di una comune bottiglia, con la base molto allargata, in cui è stato creato il vuoto (per motivi che vedremo in seguito).

Il fondo di questa bottiglia è quello che chiamiamo schermo cioè la parte che normalmente osserviamo. La sua superficie interna è rivestita con speciali materiali fosforescenti che, se colpiti da un fascio di elettroni, diventano luminescenti.

Nella parte opposta al video, cioè nel collo della bottiglia, è sistemato il dispositivo in grado di fornire il fascio di elettroni, detto "cannone elettronico".

Il fascio di elettroni può essere immaginato come un raggio di luce simile a quello che penetra da una fessura della

finestra e colpisce la parete opposta (lo schermo).

Il vuoto creato all'interno del tubo garantisce un percorso 'pulito' tra cannone e schermo, in modo che il fascio di elettroni non sia ostacolato, nel suo percorso, nè da polvere nè da gas di alcun genere (aria compresa).

Il fascio 'sparato' dal cannone, colpisce il centro dello schermo che di conseguenza si dovrebbe illuminare solo in qualche punto centrale.

Affinchè sia interamente illuminato si ricorre a dispositivi posti lateralmente al fascio, che provvedono a 'deflettere' il fascio stesso, il quale si comporta come un 'pennello' in grado di dipingere qualsiasi punto dello schermo.

Tali dispositivi sono in grado di creare campi magnetici molto intensi, tanto da poter influenzare le registrazioni dei programmi su nastro (infatti si raccomanda di tenere il registratore a distanza di sicurezza dal TV).

Un po' di tecnologia

Il compito del computer consiste, quindi, di informare tali dispositivi su come deflettere il 'pennello' e di avvisare il 'cannone' su quando sparare (o meno) il fascio di elettroni.

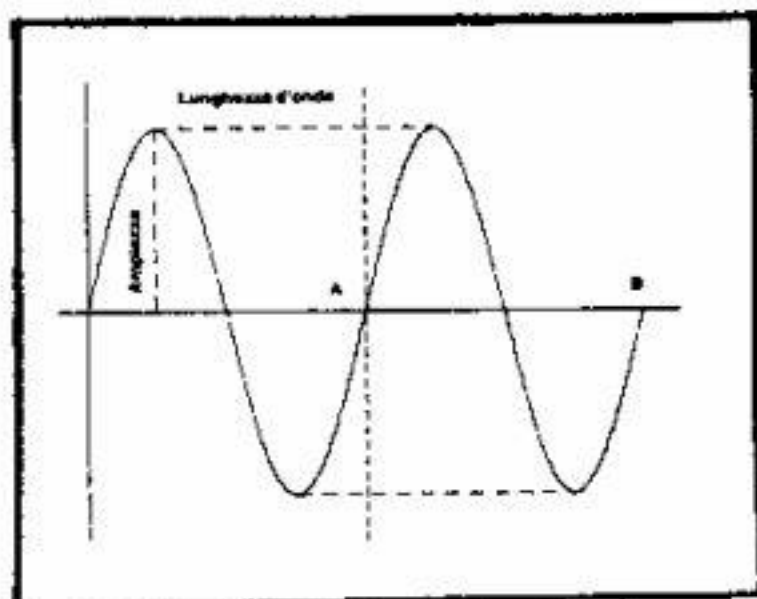



Figura A

Gli schermi, nel sistema PAL (sistema usato nei nostri televisori), sono organizzati in 625 righe. Quello che fa, in pratica, il pennello elettronico, è di effettuare una scansione dell'intero video, accendendo solo i punti che interessano. Informazione, questa, fornita dal computer.

Le scansioni, avvengono con una frequenza di 'rinfrescamento' di circa 50/60 Hertz, ogni scansione di 625 righe avviene cioè in un cinquantesimo di secondo all'incirca. Questo significa che in un secondo, il pennello elettronico ha 'ripassato' (rinfrescato) il video una cinquantina di volte.

Come mai non ce ne accorgiamo nel guardare uno schermo? Semplicemente perchè il nostro occhio gode di una proprietà, chiamata 'persistenza delle immagini': l'immagine formata sulla retina dell'occhio non scompare istantaneamente, ma vi perdura per una frazione di secondo (circa un venticinquesimo). Tutte le immagini che compaiono ai nostri occhi con una frequenza inferiore, non vengono notate.

E' proprio grazie a questa proprietà che riusciamo a vedere un film o a giocare ad un video-game avvertendo effettivamente la sensazione di movimento.

Provate ad immaginare cosa succederebbe se la persistenza delle immagini fosse pari alla frequenza di rinfrescamento. Quello che vedremmo, in un secondo, non sarebbe un movimento continuo e armonico, ma una successione di cinquanta 'quadri' statici!

Anche le comuni lampadine funzionano con correnti alternate aventi frequenze di 50 Hertz, ed è per questo che appaiono sempre 'accese'. Tutto quello visto fin'ora, vale sia per i TV in bianco e nero (monocromi) che per quelli a colori.

La differenza fra i due tipi, è dovuta solo al fatto che in quello a colori i 'cannoni elettronici' sono tre, come pure i rivestimenti dello schermo.

I tre rivestimenti producono altrettanti colori (rosso, blu, verde) ed ogni cannone si occupa della riproduzione di un solo colore. Un'opportuna combinazione dei tre colori dà luogo a tutte le sfumature necessarie per ottenere le immagini del mondo reale.

La disposizione dei tre cannoni, nel collo della bottiglia, può essere diversa a secondo della marca: possono essere disposti ai vertici di un triangolo equilatero oppure allineati. Recentemente, in qualche modello, sono stati anche riuniti in un unico cannone (sistema Sony).

Risoluzione

Per risoluzione di uno schermo si intende la capacità dello schermo stesso di separare punti (o linee) comunque vicini. In precedenza si è visto che i televisori, nel sistema PAL, sono organizzati su 625 righe. Un numero maggiore di righe migliorerebbe sicuramente la risoluzione ma, per ora, ciò non è ancora fattibile.

Quello che si può comunque verificare è che, a parità di righe, i TV monocromi hanno una risoluzione migliore di quelli a colori, che presentano un'immagine più sfuocata. La ragione è legata al fatto che, nei TV a colori, sono ben tre i pennelli elettronici da far convergere in un punto. Mentre, nei monocromi, il pennello è unico.

Da questo punto di vista, le cose vanno molto meglio per i monitor (i principi di funzionamento dei monitor a 'tubo catodico', sono gli stessi di quelli visti finora per i TV): per particolari applicazioni grafiche, possono raggiungere anche 1800 righe.

In un monitor monocromo di 19 pollici e 1000 righe, si possono ottenere punti del diametro di 0,3 mm mentre, in uno a colori, punti del diametro di 0,6 mm .

Un altro vantaggio dei monitor è dato dal fatto che il segnale del computer non deve essere 'trattato'. Per i televisori, invece, il segnale deve passare attraverso una serie di circuiti i quali introducono, inevitabilmente, un segnale di disturbo, detto genericamente 'rumore'.

Anche la dimensione dello schermo può avere una certa incidenza sulla risoluzione. Sia esso di 16 pollici oppure di 26, il numero di righe è sempre lo stesso e non è vero che uno schermo di dimensioni maggiori fornisca un'immagine migliore.

Video e computer

Vediamo brevemente come è strutturata la memoria video dei CBM 64 e C16. Sia i l C64 che il C16, suddividono il video in 25 righe per 40 colonne di caratteri: ogni 'cella' contiene un carattere, per un totale di 1000 caratteri.

Ogni cella, a sua volta, è formata da una matrice di 64 punti, distribuiti su 8 righe x 8 colonne. Il singolo punto (detto Pixel), è l'unità video fondamentale, cioè quanto di più piccolo si può "accendere" sullo schermo.

Sia sul TV di 16 pollici sia su quello di 26, il numero di punti (pixel) che si possono accendere è sempre lo stesso: la differenza è data dal fatto che su quello di 26 pollici il punto stesso avrà dimensioni maggiori, rendendo più visibile, per esempio, la struttura a gradini di una linea tracciata a 45 gradi.

Quando con un computer si opera in questo modo, cioè accendendo i singoli Pixel, si è nel modo ad 'alta risoluzione'. E' bene chiarire che per alta risoluzione del computer e risoluzione del video, si intendono due cose profondamente diverse. Quello che è implicito è che una buona risoluzione dello schermo permette una migliore visualizzazione dell'alta risoluzione del computer: il singolo pixel sarà più nitido!

Vediamo ora, computer alla mano, come ci si può rendere conto della risoluzione del nostro video con l'aiuto di qualche 'programmino' sia per il CBM 64 che per il C16.

Sono programmi che usano l'alta risoluzione: per il CBM 64 è indispensabile caricare e far girare in precedenza le solite ed insostituibili Routines Grafiche di D. Toma (CCC N.14).

Come prima prova, fate girare il programma di fig. 1: serve semplicemente a tracciare degli 'assi cartesiani' con origine nel centro dello schermo.

Osservate attentamente le due linee: normalmente, su un comune televisore (anche se "buono"!), la larghezza della linea verticale è circa il doppio di quella orizzontale. Ciò significa che la risoluzione verticale è circa la metà di quella orizzontale. Una prova più impegnativa, può essere effettuata con i programmi di figure 2 e 3. Il programma di fig. 2 traccia linee tra loro parallele nel senso orizzontale.

La riga base è data dalla linea 110;le righe in linea 120 e 130 sono poste ad una distanza di un solo pixel, mentre le ultime due ad una distanza di due pixel. Le cinque righe dovrebbero essere perfettamente distinguibili: potrebbero sorgere difficoltà solo cambiando il colore delle linee da nero a bianco (o altri colori luminosi).

Il programma di figura 3 è praticamente identico, con l'unica differenza che le cinque righe parallele sono ora verticali. Se tutte le righe sono perfettamente distinguibili, siete in possesso di un ottimo video (probabilmente un monitor monocromatico). Chi invece usa un normale televisore a colori, si accorgerà che le prime tre righe, quelle con 'passo' di un pixel, si presentano completamente fuse tra loro, non distinguibili,o, nel migliore dei casi, parzialmente distinguibili.

Può capitare che tra due televisori a colori, uno da 16 pollici ed uno da 26, si riscontri una migliore risoluzione verticale sul 26 pollici. Di quanto detto finora dovrete tener conto nel programmare i vostri lavori in 'alta risoluzione', se volete ottenere un'immagine più elegante.

Il programma di figura 4, è un esempio pratico di come tener conto della peggior 'risoluzione' verticale. Il suo scopo è di costruire un quadrato: le linee 1 10 e 130 sono i lati verticali, mentre le linee 120 e 140 quelli orizzontali.

Lo spessore dei lati verticali, è circa doppio di quelli orizzontali. Per correggere questo difetto, occorre aggiungere le seguenti linee.

Per il Commodore 64,

```
121 DRAW-50,49,0,50,49,0
141 DRAW 50,-49,0,-50,-49,0
```

Per il C 16,

```
121 DRAW1,110,51 TO 210,51
141 DRAW1,210,149 TO 110,149
```

In pratica, per avere righe orizzontali con larghezza uguale a quelle verticali, occorre 'raddoppiarle'. Per il C16, il quadrato si può costruire con la sola istruzione BOX, ma, per simmetria con il CBM 64, si è preferito il sistema descritto.

Oltre al maggior numero di righe ottenibili in un Monitor rispetto ad un TV (cosa che senza dubbio migliora la risoluzione), esiste un'altra questione tecnica riguardante la cosiddetta 'larghezza di banda' del segnale in ingresso (vedi riquadro).

Mentre un monitor può elaborare larghezze di banda anche di 10 MHZ (megahertz), un normale TV non può superare i 5.5 MHZ.

Come detto precedentemente, un TV opera con una frequenza di 'rinfrescamento' di 50 HZ. Di conseguenza, ogni punto del video a scansione, viene rinfrescato con la stessa frequenza, subendo una eccitazione di circa 20 nanosecondi (10 exp-9 sec) ad ogni scansione.

Se la larghezza di banda è bassa, il punto dello schermo potrebbe anche non essere eccitato completamente, dando luogo a differenze di intensità luminosa tra righe orizzontali e righe verticali (le righe orizzontali presentano una maggiore luminosità). Negli apparecchi a colori, potrebbe anche causare differenze di gradazioni cromatiche per uno stesso colore. Tutto sommato, il difetto risulta a volte piacevole su tali disegni geometrici!

Un altro particolare disturbo dei video a colori, è il cosiddetto 'sfarfallio'. Questo fenomeno, può essere sensibilmente ridotto aumentando la frequenza di rinfrescamento fino a 60-65 HZ, cosa realizzabile nei monitor (ricordiamo che la frequenza di rinfrescamento più usata per i TV è 50 HZ).

Questi ultimi due fenomeni (sfarfallio e variazione cromatica), sono più evidenti, a parità di qualità, su video di dimensioni maggiori.

Conclusioni

Dai discorsi precedenti possiamo trarre la facile conclusione che un monitor fornisce risultati superiori, anche di molto, a quelli di un normale TV.

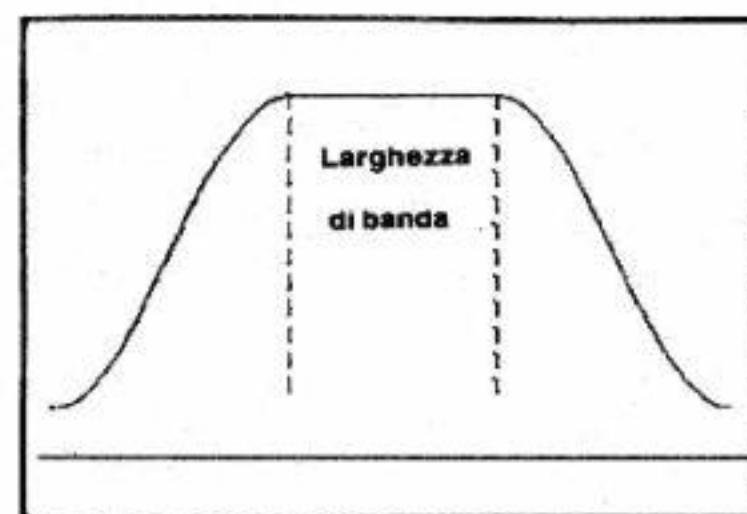



Figura B

Si può anche sottolineare il fatto che gli apparecchi monocromatici hanno risoluzioni superiori a quelle degli apparecchi a colori. Per quanto riguarda il prezzo, tra un monitor a colori o un televisore a colori delle stesse dimensioni, la differenza è minima.

La scelta tra un tipo di apparecchio o un altro, deve quindi essere fatta in rapporto all'uso che se intende fare.

Probabilmente nessuno se ne accorge, ma, nel nostro vivere quotidiano, siamo immersi in un mondo pieno di "onde". Quando ascoltiamo qualcuno parlare o cantare, o siamo infastiditi da mille rumori, la colpa non è altro che delle "onde sonore", le quali raggiungono il nostro orecchio, mettendo in vibrazione una 'membrana', che, a sua volta, produce dei 'segnali elettrici'. Questi vengono

Schema di un tubo a raggi catodici (CRT).

'decodificati' (interpretati) dal cervello, che ce li "ritorna" sotto forma di suoni (o di rumori).

Se invece ascoltate una radio o guardate un televisore, pensate per un attimo che, funzionino come il nostro cervello. Cioè ricevano delle 'onde' e, dopo una particolare decodifica, ce le trasmettano sotto forma di suoni o di immagini.

Queste onde, sono dette "onde elettromagnetiche" (formate da un campo elettrico e da uno magnetico).

Un discorso approfondito su questo tipo di onde, è davvero molto complesso: sappiate comunque che, senza di esse, non avremmo potuto vedere, tanto per fare un es., la finale Italia - Germania dei Mondiali di calcio (ahimè!).

Qualunque sia il 'tipo di onda', questa è caratterizzata dalle seguenti grandezze:

Lunghezza: distanza tra due creste successive o due successivi avvallamenti.

Frequenza: numero di oscillazioni nell'unità di tempo (secondo).

Ampiezza: distanza massima dell'onda dall'asse delle X.

La figura A, aiuterà meglio a capire queste grandezze: un'oscillazione completa, si ha nel tratto A-B.

La lunghezza d'onda è inversamente proporzionale alla frequenza dell'onda: un'onda con lunghezza molto piccola, avrà una frequenza molto alta (e viceversa).

Tornando al nostro 'televisore', qualsiasi programma è ricevuto dall'apparecchio sotto forma di onda 'elettromagnetica', che poi viene decodificata ed inviata allo schermo in modo da formare un'immagine. Ogni onda ha una sua particolare f requenza ed assolve ad un certo 'compito'.

Così, sempre nel nostro televisore, l'onda deve trasportare una informazione sull'immagine, una sul suono e, nei TV color, anche un'informazione sul colore. Perciò, non può essere composta solo da una frequenza, ma da un insieme di frequenze, detto "banda di frequenze".

Tale 'banda di frequenze' è distribuita attorno ad una frequenza detta 'portante', la quale serve per la sintonizzazione su un certo canale. Sapete benissimo che un TV può ricevere molti canali, e quindi è in grado di sintonizzarsi su diverse 'frequenze portanti'.

Succede che se il numero di informazioni che un televisore a colori deve ricevere è molto alto, ha bisogno di una Banda molto 'larga'. Quello che invece si verifica in pratica, è un compromesso tra 'quantita' e 'qualita': qualità dell'immagine da una parte e numero di canali dall'altra.

Se si raddoppiasse la 'larghezza di banda' di un televisore (da 5 MHz a 10 MHz), il miglioramento nella qualità dell'immagine sarebbe sicuramente apprezzabile, ma, per contro, si avrebbe un dimezzamento del numero dei canali ricevibili (se non si riducesse il numero di canali, le varie bande interferirebbero tra loro).

In figura B, sono mostrate due bande con larghezze differenti.

Per quanto riguarda i monitor, non essendo vincolati al numero di canali, possono raggiungere larghezze di banda sicuramente superiori. **Giancarlo Castagna**

Caratteri in Alta Risoluzione

Visualizzare caratteri alfanumerici in una pagina grafica ad alta risoluzione.

Ecco un programma didattico e un pretesto per imparare qualcosa di più sul nostro COMMODORE 64. L'argomento specifico è lo studio dell'area di memoria che risiede "dietro" lo schermo grafico ad alta risoluzione. Per fare questo useremo ancora una volta le nuove routines grafiche di Danilo Toma già pubblicate sul numero 14 di Commodore Computer Club, che dovranno essere ogni volta caricate in memoria e lanciate prima del programma in questione. E' anche consigliabile che teniate davanti a voi il numero 14 della rivista con l'articolo di Toma, che sarà molto utile per la comprensione del programma che dobbiamo studiare.

I caratteri nella Rom del 64

Prima di passare alla descrizione del programma linea per linea è preferibile accennare qualcosa sulla disposizione dei caratteri alfanumerici e semigrafici nella ROM del sistema. I caratteri del COMMODORE 64 sono "disegnati" in una matrice di otto per otto punti (comprese eventuali spaziature).

Immaginate di giocare a battaglia navale disegnando su un foglio un quadrato di otto quadretti, riempite ora i vari quadretti in modo da ottenere una lettera od un numero ed avrete così realizzato un fac-simile di uno dei 256 caratteri del vostro computer. Ogni carattere, quindi, definito da otto byte consecutivi posti uno sotto l'altro sullo schermo video.

Questi otto byte sono memorizzati nella ROM di carattere del 64 a partire dall'indirizzo decimale 53248. Per l'esattezza, da 53248 a 53255 troveremo gli otto byte corrispondenti al primo carattere (carattere " "). Da 53256 a 53263 troveremo invece gli otto byte corrispondenti al secondo carattere (carattere "A") e così via fino al 256mo carattere.

La grafica ad alta risoluzione mappata in memoria è disposta in modo compatibile con la ROM di carattere. Vediamo come: ad ogni punto luminoso sullo schermo corrisponde un bit della memoria del computer posto ad uno.

Per essere esatti, al punto luminoso in alto a sinistra sullo schermo corrispon-

dente il bit più significativo (più a sinistra) del primo byte della memoria RAM a partire dall'indirizzo decimale 57344. Al punto luminoso successivo corrisponde il bit successivo dello stesso byte 57344, e così via per i primi otto luminosi.

Il nono punto luminoso sulla stessa linea però non corrisponde più ad un bit dell'indirizzo 57344, e nemmeno ad un bit dell'indirizzo successivo (57345), ma al bit più significativo della locazione di memoria 57352, ossia otto locazioni di memoria dopo la precedente. Tutto questo perchè anche la RAM grafica è organizzata a blocchi di otto per otto punti. Un esempio: se all'indirizzo 57344 corrispondono i primi otto punti della prima linea, all'indirizzo 57345 corrispondono i primi otto punti luminosi della seconda linea e così via per le prime otto linee. Dopodichè, il byte successivo corrisponde al secondo gruppo di otto punti (dal 9 al 16) della prima linea, ed il byte seguente al secondo gruppo di otto punti della seconda linea (eccetera).

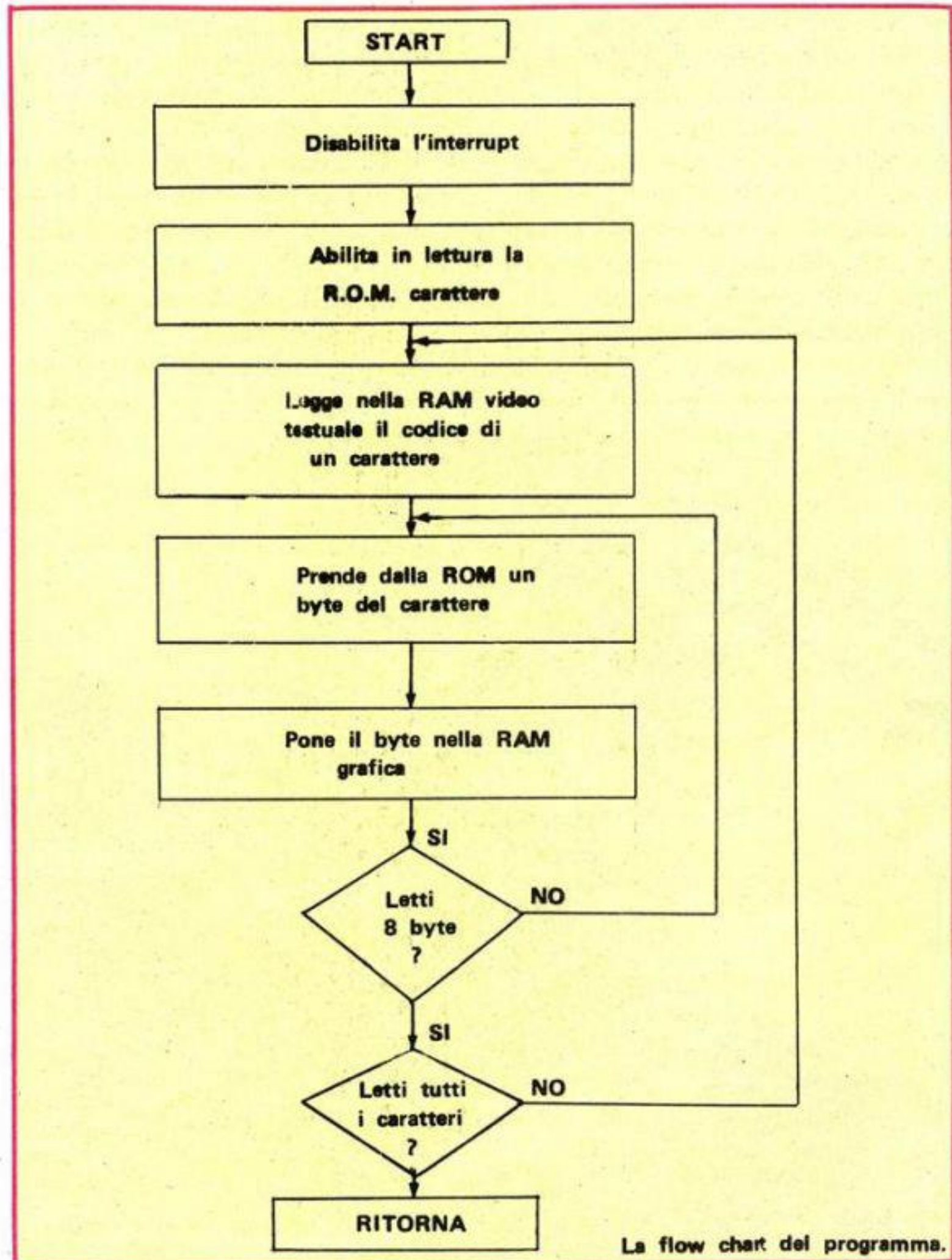
La particolare disposizione della memoria grafica permette di realizzare in modo elementare una operazione interessante: mischiare linee di testo e disegni in alta risoluzione senza ricorrere ad una complicata routine in linguaggio macchina, ma direttamente da BASIC. Questo è infatti lo scopo della subroutine che parte dalla linea 5000 e costituisce il cuore del programmino d'esempio.

Con quello che abbiamo appena detto sulla disposizione compatibile della ROM di carattere e della RAM grafica, il compito di questa subroutine è veramente semplice: leggere i codici dei caratteri presenti nella RAM di testo che appare dall'indirizzo 1024 (decimale), andare a leggere nella ROM di carattere gli otto byte corrispondenti al carattere trovato nella RAM di testo ed infine copiare questi otto byte, così come sono, nella RAM grafica.

Per utilizzare questa subroutine nei vostri programmi sarà quindi sufficiente passare al modo grafico in alta risoluzione, stampare il messaggio che volete "disegnare" con una normale istruzione PRINT, specificare il carattere di partenza nella variabile K, il numero di caratteri nella variabile J, chiamare la subroutine dalla linea 5050 e otterrete il "disegno" del vostro messaggio sullo schermo ad alta risoluzione, magari sovrapposto al disegno di un cerchio.

Carattere di partenza e numero di caratteri. Cosa significa? Che, se ad esempio dovete stampare tutta la prima linea, dovrete porre K=0 e J=40. Infatti il primo carattere è il carattere numero zero dello schermo ed il numero di caratteri che stanno sulla prima linea è quaranta.

Il programma

Passiamo ora ad analizzare linea per linea la subroutine con partenza dalla linea 5000. Alle linee 5030 e 5040 vengono definite le due variabili di cui si è già parlato sopra. La linea 5060 disabilita l'interrupt e per questo motivo non potrete fermare il programma durante la stampa dei caratteri mentre potrete fermarlo normalmente durante il disegno di

La flow chart del programma.

funzioni trigonometriche. La linea 5070 abilita in lettura gli indirizzi della ROM di caratteri che altrimenti non esisterebbe agli "occhi" del programma.

Alla linea 5090 viene definita la variabile P come base video, ossia le viene assegnato il valore dell'indirizzo di inizio della memoria RAM dedicata allo schermo testuale. Alla linea 5100 la variabile PP viene posta uguale al numero di linee grafiche che costituiscono il singolo carattere che, come abbiamo detto precedentemente è pari a otto. La linea 5110 assegna alla variabile KK il valore 53248 che come abbiamo detto corrisponde al primo indirizzo della ROM di carattere.

Nello stesso modo alla linea 5120 viene dato il valore di indirizzo della RAM grafica alla variabile JJ. La linea 5130 definisce il ciclo principale che sarà ripetuto un numero di volte pari al numero di caratteri da "disegnare" in grafica e la variabile R conterrà sempre la posizione del carattere che si sta trasferendo dallo schermo testuale a quello grafico, posizione che può andare da zero (carattere in alto a sinistra) fino a 1023 (carattere in basso a destra). La linea 5140 legge il codice del carattere dalla RAM di testo e lo assegna alla variabile CH.

La linea 5150 inizia un secondo ciclo nidificato nel primo: ha il compito di scandire verticalmente il carattere, ossia legge riga per riga il carattere dalla ROM e contemporaneamente lo scrive, sempre riga per riga, nella RAM grafica. La lettura di una riga di carattere avviene alla linea 5160, mentre la scrittura avviene alla linea 5170. La linea 5180 chiude entrambi i cicli, mentre le linee 5200 e 5220 svolgono le funzioni opposte delle 5060 e 5080, ossia la ROM di carattere torna ad essere "fantasma" mentre l'interrupt è di nuovo attivo. Attenzione a non omettere quest'ultima linea altrimenti il programma non potrà più essere fermato se non con il RESET hardware o con lo spegnimento della macchina.

La piccolissima subroutine che parte dalla linea 6000 (non utilizzata in programma) può servire per cancellare una o più linee di testo dallo schermo grafico in modo sensibilmente più veloce che non stampandovi sopra degli spazi. La routine di scrittura infatti risulta alquanto lenta perchè realizzata in BASIC. Non è comunque difficile realizzare la stessa routine in linguaggio macchina, ottenendo così una velocità comparabile (o quasi) alla normale velocità di stampa di un listato in BASIC; lascio comunque a voi il piacere di provarci.

Il resto del programma, di facile lettura e sufficientemente commentato, costituisce il main ed è volto a stampare un breve messaggio, due rette incrociate, un cerchio, una parte del set di caratteri del COMMODORE 64 ed una figura geometrica a forma di otto, in modo da evidenziare la sovrapposizione di caratteri e disegni.

Luca Galuzzi

```
100 REM  COMMODORE 64
101 :
110 REM  CARATTERI ALFANUMERICI
120 REM  IN PAGINA GRAFICA
121 :
130 REM  DI LUCA GALUZZI
190 :
200 REM  DA USARSI CON LE NUOVE RO
    UTINES
210 REM  GRAFICHE DEL N.14 DI C.C.
    C.
211 :
1110 PRINT"[CLEAR]ESEMPIO DI SCRITT
     URA IN PAGINA GRAFICA"
1120 PRINT"IL PROGRAMMA RICHIEDE L'
     USO DELLE ROUTI-";
1130 PRINT"NES GRAFICHE DI DANILO T
     OMA."
1135 PRINT"QUANDO SEI PRONTO PREMI
     UN TASTO"
1136 GET A$:IF A$="" THEN 1136
1137 ←CLEAR
1138 ←GRAF6,7
1139 ←CIRCLE0,0,0,60,50
1140 GOSUB  5000
1150 ←DRAW-160,-160,0,160,160,0
1170 ←DRAW-160,160,0,160,-160,0
1180 FOR X=1784 TO 2023
1200 POKE X,X-1784
1210 NEXT
1220 K=760
1230 J=159
1240 GOSUB 5050
1245 REM  * DISEGNA UN OTTO *
1260 FOR X=0 TO 2*π STEP .01
1270 ←PLOT100*COS(X)*SIN(X),99*SIN(
     X),0
1300 NEXT
2000 GET A$:IF A$="" THEN 2000
2010 ←TEXT 0,1
2030 END
5000 REM  *************************
     ****
5010 REM  ***   SCRIVE IN GRAFICA
     ***
5020 REM  *************************
     ****
5025 :
5030 K=0:REM  * CARATTERE DI PARTEN
     ZA *
5040 J=115:REM  * NUMERO DI CARATTE
     RI *
5050 REM  *** DISABILITA L'INTERRUP
     T ***
5060 POKE 56334,PEEK(56334) AND 254
5070 REM  * ABILITA LA ROM CARATTER
     E *
5080 POKE  1,PEEK(1)  AND  251
5090 P=1024:REM  * INIZIO RAM TESTO
     *
5100 PP=8:REM  * NUMERO LINEE PER C
     ARATTERE *
5110 KK=53248:REM  * INIZIO ROM CAR
     ATTERE *
5120 JJ=57344:REM  * INIZIO RAM GRA
     FICA *
5130 FOR  R=K  TO  J+K
5140 CH=PEEK(P+R):REM  * PRENDE UN
     CARATTERE DALLA RAM *
5150 FOR  T=1  TO  PP
5160 CG=PEEK(KK+CH*PP+(T-1)):REM  *
      PRENDE UNA LINEA DI CARATTERE
      *
5170 POKE  JJ+R*PP+(T-1),CG:REM  *
     METTE LA LINEA NELLA RAM GRAFI
     CA *
5180 NEXT T,R
5190 REM  * DISABILITA LA ROM CARAT
     TERE *
5200 POKE  1,PEEK(1)  OR  4
5210 REM  * ABILITA L'INTERRUPT *
5220 POKE  56334,PEEK(56334)  OR  1
5230 RETURN
6000 REM  *** CANCELLA PRIMA LINEA
     ***
6010 JJ=57344:REM  * INIZIO RAM GRA
     FICA *
6020 FOR  Z=0  TO  240
6030 POKE  JJ+Z,0
6040 NEXT
6050 RETURN
```

SCRIVI PIU' GRANDE SUL TUO MONITOR

di Fabio Sorgato

Il programma è talmente breve che vale la pena di digitarlo, almeno per imparare qualcosa di nuovo. Il suo compito consiste nel creare messaggi in doppia altezza utilizzando la tecnica dei caratteri ridefinibili, cioè programmabili a piacere. Il nucleo del listato consta di due subroutines, una per attivare e l'altra per disattivare i caratteri in doppia altezza.

Prima di far 'partire' il programma sarebbe consigliabile digitare:
POKE55,0:POKE56,48:CLR
per evitare problemi come spiegato nella seconda parte dell'articolo.

Le scritte raddoppiate si attivano con:
GOSUB 63000
e si possono controllare varie funzioni per mezzo di parametri (=variabili).

A$: serve a contenere il messaggio (non più di 127 caratteri).

Come creare caratteri in doppia altezza riproducendo la possibilità offerta dal Vic 20.

XP,YP: rappresentano le coordinate x e y dello schermo in cui inizieranno le scritte. I limiti per X ed Y sono:

0 <= X <= 39
0 <= y <= 25

CO: colore delle scritte secondo la tabella standard.

MI: se MI=1 le scritte da stampare saranno in minuscolo/maiuscolo, se diverso da 1 saranno in maiuscolo/segni grafici.

AD: se AD=1 le scritte verranno stampate senza cancellare messaggi già presenti sullo schermo. In caso contrario (AD diverso da 1), lo schermo sarà cancellato.

Con GOSUB 63500 si disattivano le scritte in doppia altezza e si ritorna alle scritte normali.

La prima parte del listato rappresenta una piccola dimostrazione delle varie funzioni del programma.

Come funziona

Quando accendiamo il Commodore 64, è attivato automaticamente il modo di caratteri standard, cioè quello che normalmente usiamo per programmare.

E' noto, però, che i caratteri possono essere prelevati sia dalla RAM che dalla ROM e per creare dei nuovi caratteri sarà sufficiente segnalare al Vic II (circuito integrato che si occupa della gestione del video) di prelevare i caratteri dalla RAM.

Ogni carattere può essere contenuto in una griglia 8x8 dove ogni singolo punto (detto pixel) può essere 'acceso' (visibile) o 'spento' (invisibile).

Ogni set di caratteri occupa 2000 byte (2K), dato che ogni carattere è formato da 8 bytes ed un set completo è formato da 256 caratteri (128 in "normale" e 128 in reverse). Poichè possiamo disporre di due set di caratteri (minuscolo/maiuscolo e maiuscolo/semigrafico) il generatore di caratteri nella ROM occupa 4K di cui è però possibile utilizzare 256 caratteri alla volta.

Come accennato in precedenza per costruire un nuovo set di caratteri sarà sufficiente segnalare al Vic II la nuova partenza in RAM e a scopo sperimentale useremo la locazione 12288.

Per entrare nel nuovo set sarà sufficiente eseguire:
POKE53272,(PEEK(53272)AND240)
+12

mentre per tornare alla visualizzazione normale sarà sufficiente premere i tasti RUN/STOP e RESTORE.

Proviamo ora a creare nuovi caratteri grafici. Innanzitutto per proteggere il nuovo set di caratteri da eventuali sovrascritture si deve ridurre la memoria. Semplicemente si abbassano i puntatori di fine programma Basic in modo che i programmi che scriveremo non andranno a sovrapporsi con il nuovo set. Per ritornare alle condizioni di memoria iniziale digitare: SYS64738 (attenzione: in questo modo viene cancellato qualsiasi programma Basic presente in memoria.)

Per proteggere il set digitiamo:
POKE 55,0:POKE56,48:CLR

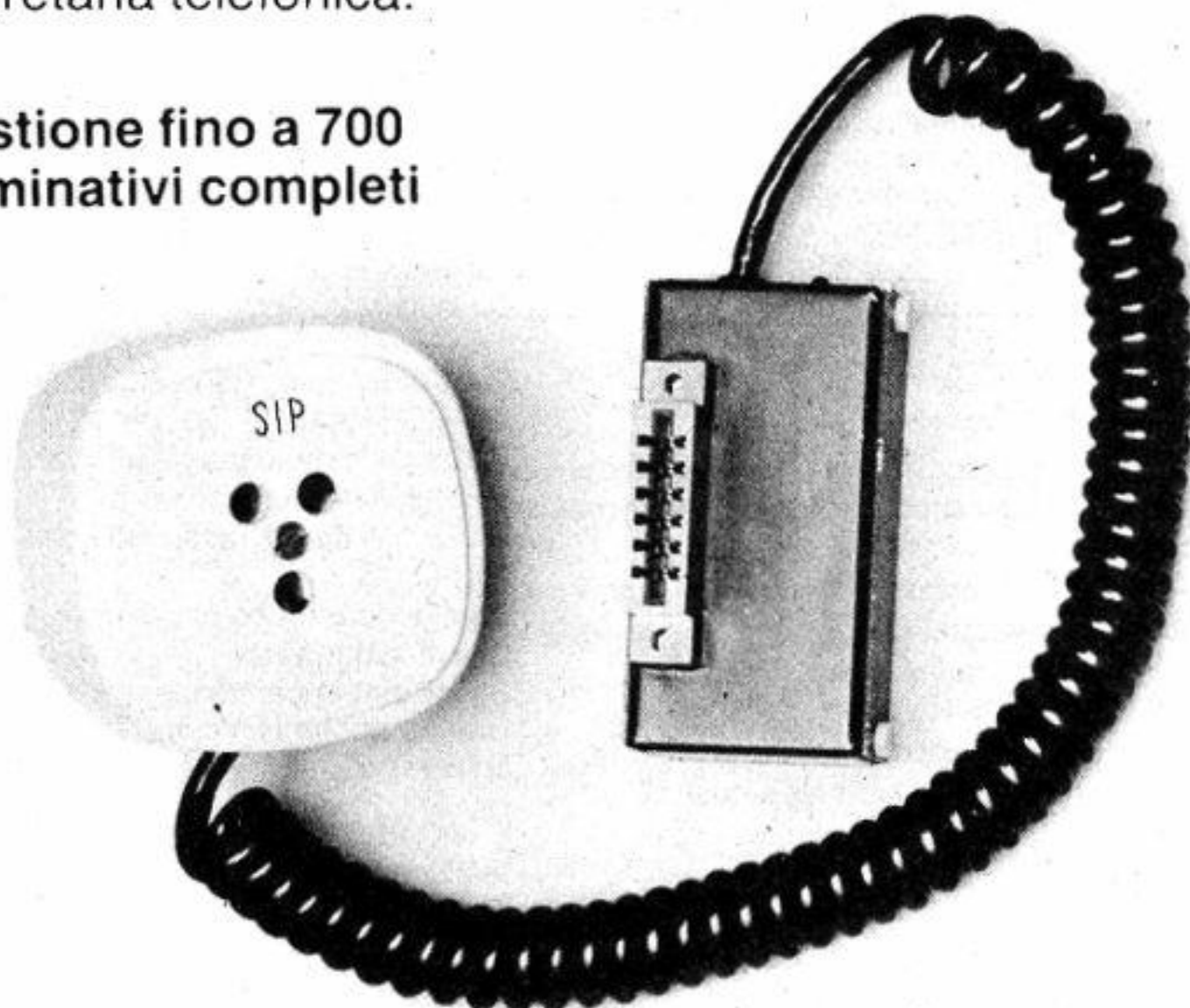
Ora digitiamo il programmino di prova.
Commento:

LINEA 5 passa al maiuscolo
LINEA 10 riserva la memoria per i caratteri
LINEA 20 disabilita l'interrupt
LINEA 30 switch dei caratteri
LINEA 40 carica i caratteri
LINEA 50 switch I/O
LINEA 60 riabilita l'intterrupt

Ora eseguiamo:

POKE 53272,(PEEK(53272)AND240)
+12

Non accade nulla di speciale. Perchè?

Il CBM 64 sta prendendo i caratteri dalla RAM dove però risiedono i caratteri copiati dalla ROM con il programma.

Scriviamo la (A commerciale) e muoviamoci con il cursore di due spazi più in basso. Scriviamo:

FORI=12288TO12288+7:POKEI,
255-PEEK(I):NEXT

Il carattere sarà sempre lo stesso ma visualizzato in reverse. Per ritore alla normalità basterà dare il RUN al programma.

Cancelliamo ora (NEW) e scriviamo il seguente programmino:

```
10 FORI= 12448TO12455:
READA:POKEI,A:NEXT
20 DATA 60,66,165,129,165,153,66,60
```

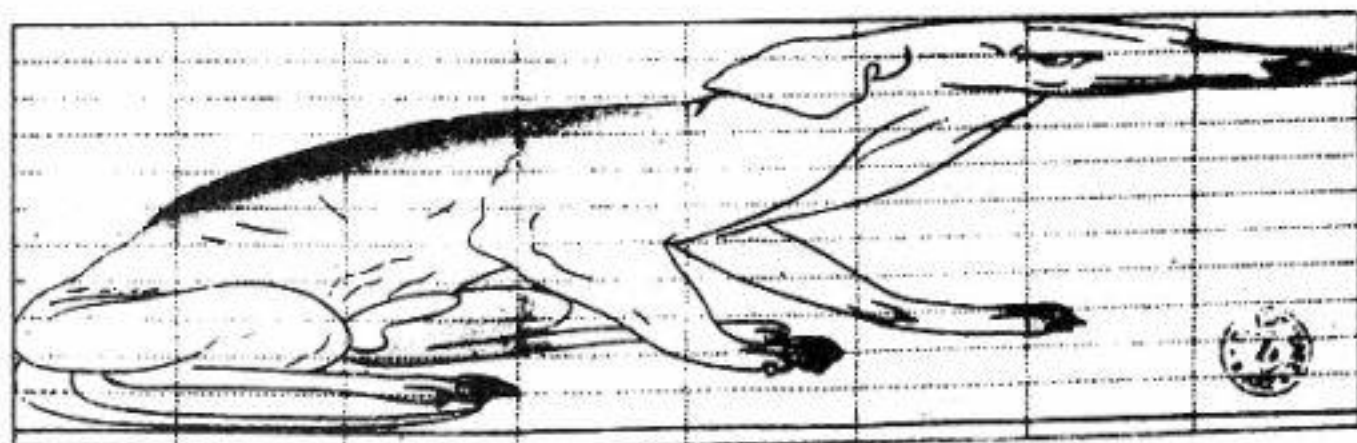
Dando il RUN, e premendo il carattere T, avremo un nuovo disegno.

La tecnica per creare nuovi caratteri è di disegnare su di un foglio a quadretti un riquadro di 8x8 e di annerire ogni singolo quadretto per rendere visibile un punto.

Terminato il disegno, convertire le singole righe da binario in decimale.

La formula per trovare la locazione di partenza del carattere da ridefinire è uguale a:

START+8 * COD

in cui: START è la locazione del generatore di caratteri (in questo caso 12288) COD il numero del codice della memoria di schermo del carattere.

Le immagini di questo fascicolo

Si trovano a Londra queste stampe di cane. Prima normale, poi allungato, ma sempre secondo precise regole geometriche. Da notare la quadrettatura che aveva la funzione di creare i riferimenti per il disegnatore.

```
 8 REM   COMMODORE 64
 9 REM   CARATTERI IN
10 REM  DOPPIA ALTEZZA
11 REM  BY FABIO SORGATO
12 REM  MEDA (MI)
13 :
14 :
15 POKE 56,48:POKE 55,0:CLR :REM
    FISSA TOP DI MEMORIA
20 A$="HO LE MAIUSCOLE":AD=0:XP=1
   2:YP=10:CO=0:GOSUB 63000:GOSUB
    150
30 A$="♥♥ CHE POSSO USARE ♥♥":AD=
   0:XP=10:YP=8:CO=0:MI=0:GOSUB 6
   3000:GOSUB 150
31 :
40 A$="CONTEMPORANEAMENTE CON LE
   MINUSCOLE !!!!"
50 AD=1:XP=0:YP=13:CO=0:MI=1:GOSU
   B 63000:GOSUB 150
60 MI=0:CO=0:AD=0:FOR YP=2 TO 10
   STEP 2:A$="●●● HO TUTTI I COLO
   RI"
70 CO=CO+1:XP=XP+1:GOSUB 63000:AD
   =1:NEXT:GOSUB 150:GOSUB 63500:
   END
  150 FOR T=1 TO 3000:NEXT:RETURN
  160 :
  170 :
62999 REM  INIZIO SUBROUTINE
63000 X=12288:REM  INDIRIZZO NUOVA M
      APPA CARATTERI
63010 MI=(-(MI=1))*2048:REM  MINUSCO
      LE
63020 IF A$="" OR  XP>40  OR  XP<0
      OR  YP>25 OR YP<0 OR XP+YP*40+
      LEN(A$)>1000 THEN RETURN
63030 CO=CO  AND  15
63040 A=53248:REM  INIZIO ROM CARATT
      ERI
63050 POKE  A+24,29:REM  SELEZIONA N
      UOVA MAPPA CARATTERI
63060 IF AD=1 THEN LA=LA+LEN(A$)*2:G
      OTO 63130
63070 PRINT CHR$(147);
63080 FOR K=1 TO 25
63090 PRINT"[BLEU]
                                [CELESTE
      ]";:REM  40 SPAZI
63100 NEXT
63110 PRINT CHR$(19)
63120 LA=0
```

```
63130 A1=LA
63140 IF LA>=255 THEN LA=0:RETURN
63150 A2=LEN(A$)+LA
63154 REM  CREA SPAZIO PER MESSAGGIO
63155 FOR KK=X+A1*8 TO X+LEN(A$)*16+
      A1*8:POKE KK,0:NEXT
63160 IF AD=0 THEN LA=LEN(A$)*2
63170 FOR I=0 TO LEN(A$)-1
63180 K=I+XP+YP*40+40*INT((I+XP)/40)
63190 POKE 1024+K,I+A1:REM  PRIMA ME
      TA' NUOVI CARATTERI IN MEMORIA
       DI SCHERMO
63200 POKE 1064+K,I+A2:REM  SECONDA
      META' NUOVI CARATTERI IN MEMOR
      IA DI SCHERMO
63210 POKE 55296+K,CO:REM  COLORE
63220 POKE 55296+40+K,CO:REM  COLORE
63230 NEXT
63240 POKE 56334,0:POKE 1,51:REM  TO
      GLIE INTERRUPT E I/O
63250 D=X
63260 FOR I=1 TO  LEN(A$)
63270 S=ASC(MID$(A$,I,1))-64
63280 IF S<0 THEN S=S+64
63290 IF S>128 THEN S=S-64
63295 REM  CREA CARATTERI IN DOPPIA
      ALTEZZA
63300 FOR J=0 TO 7
63310 M=PEEK(53248+(S*8)+J+MI)
63320 FOR K=0 TO 1
63330 IF J<=3 THEN POKE D+A1*8,M:D=D
      +1
63340 IF J>3 THEN POKE D+A2*8-8,M:D=
      D+1:GG=GG+1
63350 NEXTK
63360 NEXTJ
63370 D=D-8
63380 NEXTI
63390 POKE 1,55:POKE 56334,1:REM  RI
      PRISTINA I/O E INTERRUPT
63400 RETURN
63500 POKE 53272,21:PRINT CHR$(147):
      RETURN
63510 REM  RIPRISTINA SET STANDARD C
      ARATTERI
```

SEVEN CITIES OF GOLD

Se avete sempre desiderato scoprire nuove terre e conquistarle per diventare un eroe come Cristoforo Colombo, Amerigo Vespucci, Pizarro e tanti altri questa è l'occasione che fa per voi.

Questa stupenda avventura scritta da Ozark Softscape (fratelli Bunten e amici, che hanno già realizzato il bellissimo M.U.L.E., sempre della Electronic Arts), comprende un game disk, un manuale di istruzioni e di una reference card per una veloce consultazione.

Per avvicinare il più possibile il giocatore alla realtà, il programma richiede la creazione di un data disk dove il computer utilizzando dati geografici storici e demografici provvederà a creare terre, montagne, fiumi, villaggi e abitanti. Dopo una breve attesa di circa venti minuti potremo finalmente iniziare i nostri viaggi nei mondi sconosciuti.

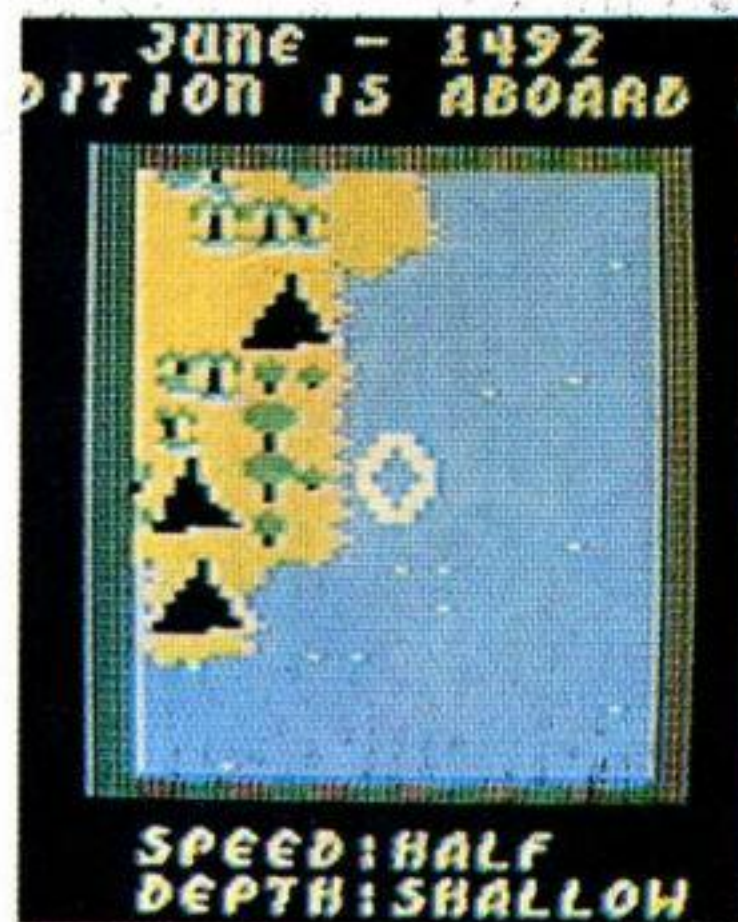
Dopo una schermata introduttiva (venite a conoscenza del nuovo mondo) la corte vi nominerà capitano a capo di una spedizione composta da 4 navi, 100 uomini, cibo e merci di scambio necessarie per circa due anni. Vi troverete nel marzo del 1492 in Spagna davanti al palazzo del re.

Potrete scegliere se entrare nel palazzo o, muovendovi verso sinistra, visitare il pub dove avrete la possibità di salvare su disco la vostra avventura, o se andare in casa dove avrete la situazione delle terre, fiumi, montagne, nativi e oro conquistate fino a quel momento con a fianco la mappa dei territori visitati. Volendo potrete cambiare la quantità delle vostre provvigioni entrando nell'"outfitter";qui potrete comprare nuove navi, uomini, cibo e merce variando le proporzioni che la corte ha definito per voi; aspettate però di avere già una buona esperienza di navigazione per affrontare queste decisioni. Di solito la corte stabilisce delle proporzioni eque.

Per orientarvi tenete conto che Nord è l'alto dello schermo, Sud è il basso, Est è la destra e Ovest la sinistra e la vostra patria è 30 gradi di latitudine Nord. Questo vi sarà molto utile per tornare a casa.

Una volta imbarcati vi si presenterà una schermata con al centro una finestra che mostra la vostra flotta mentre solcate le onde dell'Oceano ed ai lati la situazione delle navi, del cibo, degli uomini e della merce. Muovendo la vela del joystick inserito in porta due avrete la possibilità di muovervi nella direzione scelta; premendo il pulsante potrete scegliere di vedere la mappa di navigazione, sbarcate su una nuova terra o riprendere il viaggio.

A Nord e a Sud troverete le terre polari dove il vostro equipaggio si rifiuterà di sbarcare. Sulla linea dell'equatore troverete popoli civilizzati o tribù di cacciatori. A Ovest potrete cercare nuovi mondi per soddisfare le vostre aspirazioni. Durante il vostro viaggio potrete incontrare anche violenti temporali.

Una volta arrivati a terra potrete decidere con quanti uomini, cibo e merce di scambio andare in esplorazione. Potrete esplorare villaggi, montagne e scoprire nuovi fiumi. Una volta a terra, dovrete stare molto attenti ai messaggi di pace che il computer vi comunicherà, per evitare di cadere in imboscate ad opera dei nativi. E' una ottima idea portare con se molta merce per tranquillizzare le ire degli

abitanti ed uscire dalle situazioni più difficili e pericolose.

Entrati nel villaggio, premendo il pulsante potrete donare merce ai nativi, cercate di fare commercio guadagnando oro e cibo, attaccarli per derubarli o riprendere l'esplorazione. Ricordatevi, però, che la corte terrà conto del vostro buon comportamento per ricompensarvi con titoli e cariche onorifiche come quelle di Governatore e Vicerè.

Potrete decidere di concludere la vostra avventura, se riuscirete a sopravvivere, con il ritorno a casa. Per tornare a casa ricordatevi di seguire la direzione dell'inizio; una volta arrivati potrete rientrare a casa per vedere la vostra situazione, salvare su disco la situazione attuale per riprenderla in seguito, andare alla corte per ricevere onorificenze e nuovi soldi per ripartire alla scoperta di nuove terre.

Il gioco permette anche di fare una copia del data disk per far si che due giocatori si misurino sulla stessa mappa creata dal computer (il computer tiene conto dei fatti storici, ma in modo casuale).

Idea:	7
Giocabilità:	9
Grafica e suono:	9
Animazione:	0
Voto:	9

Spooks1

Questo gioco, creato dalla Mastertronic, rientra nel filone dell' orrore cui sono stati dedicati già alcuni games sul Commodore 64.

La casa della defunta e stravagante prozia è piena di spettri ed esseri demoniaci.

Vostro obiettivo è quello di individuarli e di esorcizzarli.

Ecco due consigli per agevolarvi il compito.

In primo luogo dovete cercare di spostare le lancette di ogni orologio presente nella casa su un orario successivo alla mezzanotte. Se non riusciste, sareste immancabilmente perseguitati dagli spettri presenti nell'appartamento.

Inoltre è necessario che voi ricomponiate la melodia composta da 8 brani, nascosti in altrettanti punti.

Quando suonerete questa marcia funebre, i fantasmi verranno esorcizzati e scompariranno per sempre dalla vostra abitazione.

Buon divertimento e non abbiate paura di vedere ciò che si trova oltre la porta!!!.

Idea	9
Giocabilità e suono	8
Animazione	7
Voto	8

R.I.P.

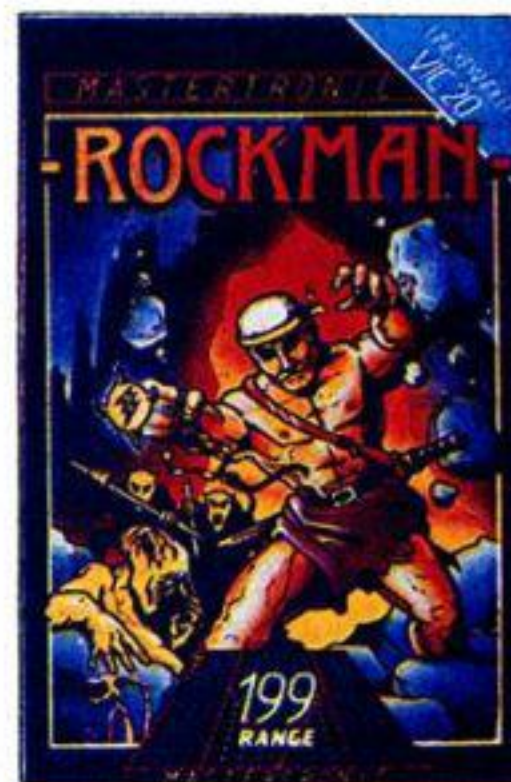
R. I.P., ovvero Riposa In Pace, è il nuovo gioco della Mastertronic che vi condurrà nel mondo dell'occulto. In un paese incantato sono stati sottratti al Re venti calici della Verità.

Per impedirire al Male di regnare indisturbato bisogna riconsegnarli alla casa reale. Il compito non è semplice dato che sono stati portati nelle cripte delle Tenebre dove creature assetate di sangue li difendono fino alla morte.

Il filo conduttore che collega lo scopo del gioco al suo titolo è proprio questo: armati con la Bacchetta della Luce ci innoltriamo nelle Tenebre e, oltre a prendere i calici, dobbiamo cercare di riportare a naturale morte le orrende creature.

Per totalizzare quindi il maggior punteggio dobbiamo, in cinque vite, raccogliere più calici possibili evitando di perderci nel labirinto delle Tenebre.

Idea	8
Giocabilità e suono	9
Animazione	8
Voto	8

Rockman

Questo nuovo prodotto della Mastertronic racchiude in sè due caratteristiche di gioco molto avvincenti che ne fanno un oggetto per veri intenditori dell'avventura.

La prima è l'affascinante sensazione che si prova nel trovarsi all'interno di un labirinto diviso in stanze collegate una con l'altra. Una volta entrati, si deve stare molto attenti all'orientamento: solo coloro che non si lasceranno prendere

dal panico e dalla fretta riusciranno a muoversi nel dedalo di stanze.

La seconda caratteristica è dettata dalla mancanza di una vera e propria arma a nostra disposizione. Non avremo perciò pistole laser o capacità sovrumane per difenderci lungo il percorso, ma dovremo affidarci alla logica per spostare, nella maniera opportuna, delle rocce. Queste saranno utili a volte per schiacciare i nemici presenti nelle stanze del labirinto, a volte per poter raccogliere i 160 pezzi di un amuleto magico.

Dopo averli recuperati tutti avremo portato a termine la nostra missione.

Idea	7
Giocabilità	8
Animazione	8
Voto	8

Skyjet

Quando decollerete con il vostro elicottero militare, vi attenderà una missione densa di pericoli e incertezze.

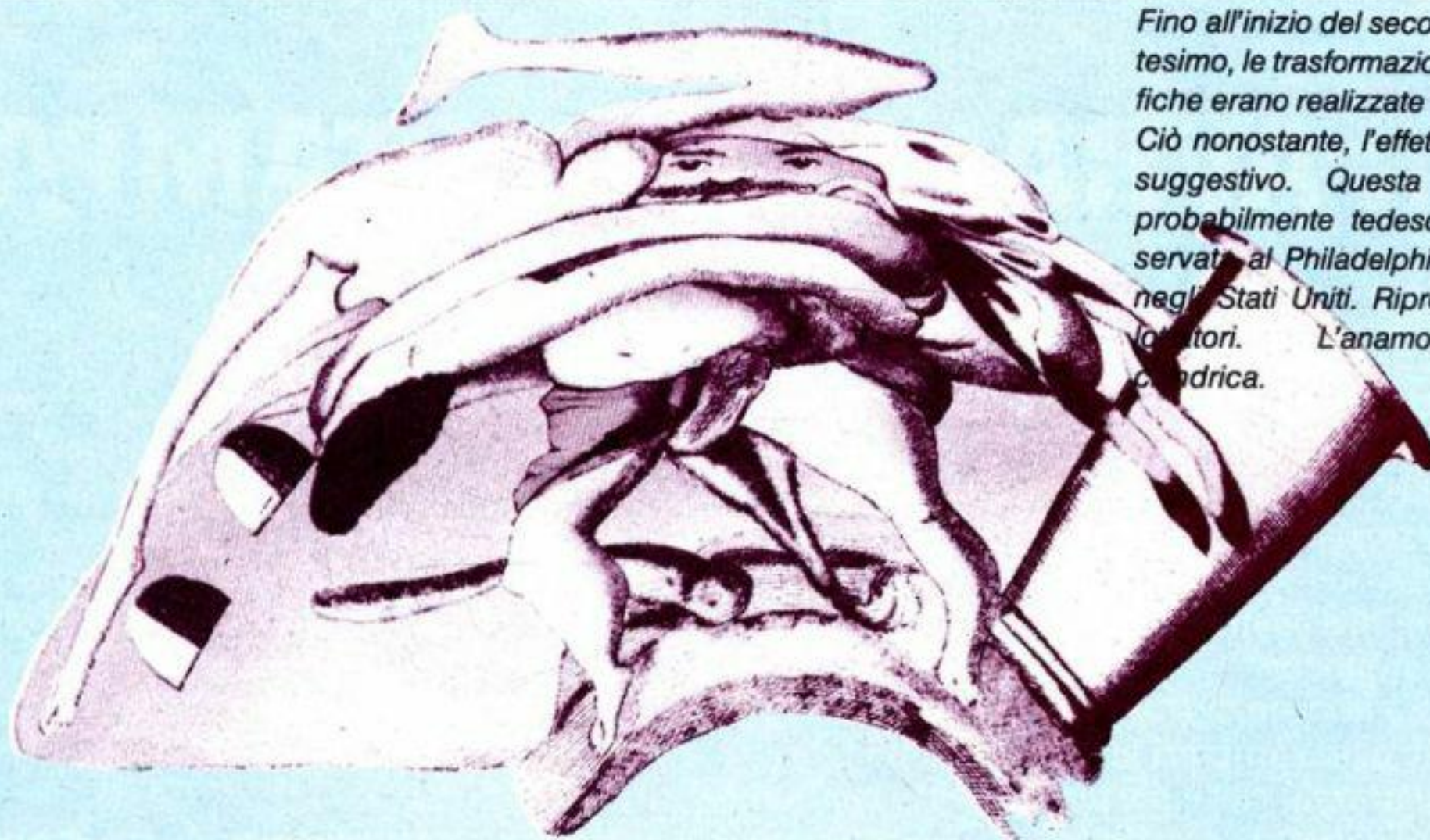
Delle navi nemiche tenteranno di abbattervi con la loro artiglieria di precisione.

Le squadriglie aeree avversarie cercheranno di impedire la vostra incursione in territorio nemico, prendendovi di mira con i loro missili di straordinaria potenza distruttiva.

Inizialmente potrete selezionare l'intensità dei rumori di fondo dello scontro aereo.

Questo gioco della Mastertronic introduce degli interessanti elementi di novità nella ormai vasta e gradita serie dei giochi di simulazione sul computer.

Gli amanti di questo genere di game trarranno indubbiamente soddisfazione da questo nuovo prodotto.

Idea	7
Giocabilità e suono	8
Animazione	7
Voto	7

Fino all'inizio del secolo diciasettesimo, le trasformazioni anamorfiche erano realizzate "a tentoni". Ciò nonostante, l'effetto risultava suggestivo. Questa incisione, probabilmente tedesca, è conservata al Philadelphia Museum negli Stati Uniti. Riprodotti, due lottatori. L'anamorfosi è cilindrica.

Vendo per Commodore 64 tavoletta grafica suncom "Animation Station" quasi mai usata completa di software su disco e manuale istruzioni a L. 170.000. (Andrea Haardt - Via Bellincioni, 20 20097 San Donato Milanese - Tel. 02/5279360).

Vendo Vic 20 con registratore C2N, 60 giochi, 2 cartridge, libro BASIC 2, joystick, paddles, manuale di istruzioni in italiano. L. 360.000. (Sebastiano Soriani - Via del Lavoro, 1 44100 Ferrara - Tel. 0532/51261).

Vendo Vic 20 completo di imballo, manuale giochi a L. 90.000. (Gianluca Caloti - Via Einaudi, 33/B - 40132 Bologna - Tel. 051/568754).

Vendo Vic 20, registratore originale, espansione 3-8-16K, joystick, paddles, 8 cartridge origianle Commodore, 60 giochi, 15 lezioni su cassetta "Conoscere il computer duirettamente dal computer", 10 programmi utility. L. 380.000 trattabili. (Antonio Lettini - Corso Vercelli, 9 - 20110 Milano - Tel. 02/494873).

Vendo Vic 20, manuali tradotti, programmi in cassetta (utility, giochi) interfaccia registratore. (Dino Bonanzinga - Via Torr. Trapani, 13 98100 Messina - Tel. 43439).

Vendo Vic 20, registratore originale, 16K, cartridge Assembly Mes Mon con utilty (Word processor, 40 colonne, ecc.) e se giochi (Sargent II, Solar System, Pac Man, Jet Pack ecc.) per un totale di oltre 90 programmi. Il tutto a L. 350.000 (Ricardo De Mattia - Via Rinuccini, 3 - 20125 Milano - Tel. 02/2070233).

Vendo libri Vic 20 e BASIC nuovissimi a buon prezzo. (Francesco Versace - Rione Guarna, 70 - 89100 Reggio Calabria).

Vendo Vic 20, registratore a L. 200.000 (Agostino Vanzino - Via G. Marconi, 16 - 14055 Costigliole d'Asti (AT) - Tel. 0141/966772).

Vendo Computer Commodore 8032, Floppy 8050 1Mb, stampante 8024 160 Cps, vasta libreria di software. Prezzo da concordare. (Marco Succi - Via Bartolini, 146 48100 Ravenna - Tel. 055/38000).

Vendo stampante Commodore MPS 801 a matrice di puntim, grafica, 80 colonne, praticamente nuova, con imballo, manuale completa di software dedicato L. 350.000 (Luca Bollati-Guzzo - Via Renato Fucini, 43 - 00137 Roma - Tel. 06/8184472).

Vendo a L. 75.000 registratore C2N Commodore mai usato, per sbagliato acquisto. (Maurizio Lupi - Via della Libertà, 71 - 50050 Vitolini - Tel. 0571/584218).

Vendo Vic 20, libro del Commodore Vic 20. (Lillo Marino - Via Serradifalco, 113 - 90145 Palermo - Tel. 091/577616).

Vendo Vic 20 con espansione da 28K con inserita RAM con programmi TTTY (Bandot, ASCII) Morse (CW) L. 300.000 (Mario Pellegrini - V.le Fulvio Testi, 200 - 20092 Cinisello Balsamo (Mi) - Tel. 02/2405172).

Vendo Vic 20, registratore C2N, Superexpander, Hes Mon, 2 cartridge, jojstick, 4 libri e molti giochi in cassetta. Tutto a L.300.000 (Roberto Stanghellini - Via Italia, 40 - 20094 Corsico (Milano) - Tel. 02/4583257).

Vendo CBM 64, Floppy disk, stampante, joystick, neumerosi programmi, utility, imballi, originali, ancora in garanzia. (Francesco Forte - Via Cairoli, 14 27029 Vigevano - Tel. 0381/79170).

Vendo Vic 20, joystick, svariati programmi su cassetta, 3 cartucce, varie riviste al prezzo di L. 160.000. Tre mesi di vita, come nuovo, con imballaggio originale e un manuale in italiano. (Cristian Riviera - Via Lasagna, 6 - 47035 Gambettola (FO) - Tel. 0547/52473).

Vendo Vic 20, registratore, joystick, 3 cartucce, circa 40 programmi su cassetta, il tutto a L. 350.000. (Luca Arbore - Via Boggio, 2 - 10015 Ivrea (TO) - Tel. 48722).

Vendo Commodore CBM 8032 da 1 Mbyte, stampante 3022, registratore C2N, interfaccia IEEE/RS-232, compilatore Basic, più molti linguaggi, programmi, manuali ed accessori vari a L. 4.900.000 trattabili. (Giuliano Giannini - Via Tartaglia, 7 - 20154 Milano - Tel. 02/382797).

Vendo sistema Vic 20 con espansioni, libri vari, programmi su cassette già registrate a prezzi di svendita. (Mario Novelli - Via Cavour, 15 - 21013 Gallarate (VA) - Tel. 0331/786474).

Vendo Vic 20, Superexpander, pubblicazioni e cassette per un valore di L. 90.000 (Antonio Solinas - Via Cellini, 1 - 09100 Cagliari - Tel. 43114).

Vendo per Vic 20 Cartridge espanso 16K RAM. (Ugo Vassallo - Corso Martiri della Libertà, 14 - 95131 Catania - Tel. 326960).

Vendo Vic 20, espansione 8K RAM, 3 cartridge, 11 cassette di programmi e play games, 2 cassette con oltre 70 programmi, "Introduzuone al Basic 1 e 2" con 4 cassette e i numeri 6/7/9/10/11/12/13/15 di CCC. Il tutto a L. 400.000 trattabilissime. Vendo anche separatamente. (Paolo Serafini - Via G.B Gandini, 21 - 00167 Roma - Tel. 06/6376370).

Vendo Vic 20 e registratore C2N con manuali di istruzione e imballi originali, 300 videogiochi, 3 libri con vari listati. Il tutto a L. 350.000 (Mauro Farina - Via Masotto, 30 - 20133 Milano - Tel. 02/7380112).

Cerco Commodore 64 in buono stato. (Raineri Antonio - Via Mazzini, 1 - 20077 Melegnano (MI) - Tel. 02/4833788).

Vendo Vic 20, registratore dedicato, numerose riviste, nemerosi listati, joystick, acquistati da neanche due mesi. Il tutto ad un prezzo stracciatissimo. (Franco Toma - Via Scorrano, 84 - 73024 Maglie (LE) Tel. 0936/25529).

Vendo Vic 20 usato pochissimo completo di cavetti e alimentatore, espansione 3-816K, software consistente in 41 programmi su nastro, 3 cartucce originali, circa 90 listati a L. 350.000. Non trattabili. (Franco Nucci - Via Antonio Locatelli, 13 - 24032 Calolziocorte (BG) - Tel. 0341/644944).

Offro consulenza. Dimostratore per computer e periferiche Commodore in negozi specializzati. Solo per la zona Vomero. (Enrico Antinozzi - Corso Europa, 26 - 80127 Napoli).

Offro consulenza. Zona Treviso-Pordenone-Udine offresi dimostratore e/o installatore "Commodore" part-time dal lunedì al sabato. Esperto programmazione/analisi (Cobol Honeywell"DPS 4"). Adeguate conoscenze programmazione BASIC, Commodore C64). Mattine o pomeriggi liberi nel periodo indicato. (Carmine Cuppone - Via Spellanzon, 80 - 31015 Conegliano Veneto (TV) - tel. 0438/62011).

Offro consulenza. Preparo legge 373, calcolo dispersioni termiche edificio professionale personalizzata in Basic o linguaggio macchina per Commodore 64 o PLus/4. (Giorgio - Tel. 02/2845265).

Offro consulenza. Offro collaborazione per programmi e articoli in particolare su argomenti matematici, scientifici e di utilità. (Flavio Molinari - Via Friuli, 38 - 20020 Lainate (Milano) - Tel. 02/93257030).

Cerco possessori Commodore 64 e drive. (Michele Loschi - Via Zanini - Cornuda (TV) - Tel. 042383164).

Cerco hobbisti possessori di Commodore 64. (Alessandro Zucchi - Località Tirafiume - 28052 Cannobbio - Tel. 0323/70835).

Cerco unità Floppy Dysk drive in buone condizioni. Inoltre vendo e cambio giochi. Cad. L. 6.000 (Massimo Zuffi - via Sapri, 2 - 44100 Ferrara - Tel. 0532/96608).

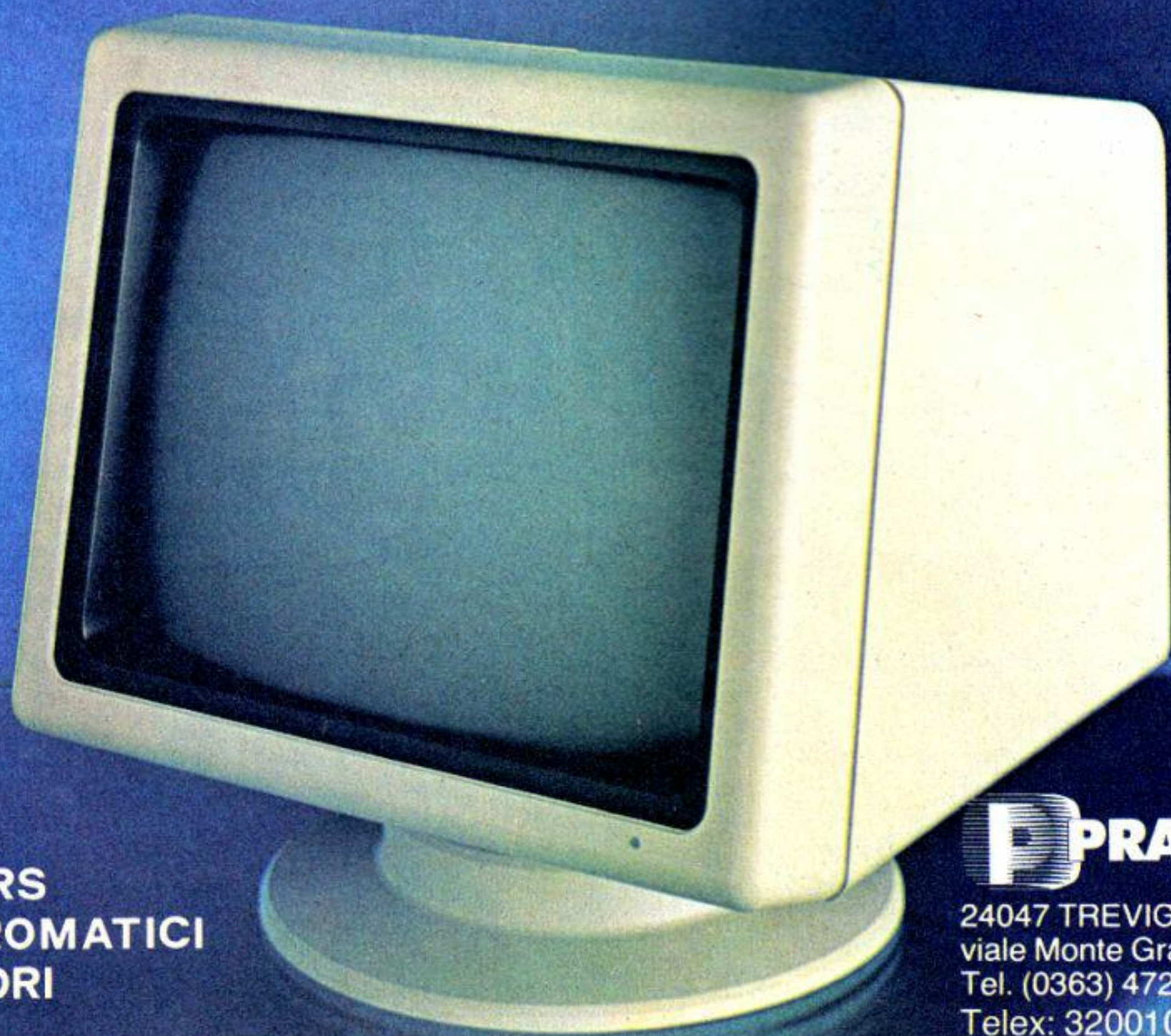
Cerco stampante usata ma funzionante ad un prezzo modico. Inoltre cerco cartridge Basic 4.0 con manuale ad un prezzo trattabile. (Cristoforo Marcosanti - Via Resistenza, 90 - 11026 Pont St. Martin - Tel. 0125/84750).

Cerco Commodore 64 con drive 1541 e stampante ed eventuali copie dei numeri 1 e 2. (Gennaro Esposito - Via S. Gervasio, 39 - 50131 Firenze - Tel. 055/572753).

Cerco registratore C16, programmi su carta per C16, manualetto C16 delle istruzioni in italiano oppure appunti. (Giovanni Lazzaroni Via C. Albani, 8 - 27100 Pavia - Tel. 461055).

Acquisto fotocopia del manuale G-Pascal 3.0 della Gambit - Games per C64. (Sandro Crocerossa - Via della Pace, 1 - 87040 Castrolibero (CS)).

Cerco i seguenti programmi su nastro: vero simulatore di volo, Sam reciter, totocalcio (elaborazione sistemi). Solo se prezzi modici. (Ernesto Orga - Via Boezio, 59 - 80124 Napoli Tel. 081/322803).

Cerco istruzioni del programma "S.A.M." del C64. (Paolo Ventaglio - Via angelo Olivieri, 55A/1 - 16133 Genova - Tel. 010/381182 casa - 010/322803 ufficio).

SCAMBIATEVI LE LISTE o promuovete un Club

Stefano Dominioni - Via Niccolo' Tommaseo 18 - 21100 Varese - Tel. 0332/229909

Franco Piccinno - V.le Leonardo da Vinci 146 00145 Roma - Tel. 06/5132153

Massimiliano Campili - Via Carnia 8/A - 05100 Terni - Tel. 0744/59497

Micro Commodore Club - Via Panizzi 13 - 20146 Milano - Tel. 02/427890

Silvio Pazzaglia - Via Marconi 85 - 37060 Castel d'Azzano (Vr) - Tel. 045/512340

Vincenzo Arcidiacono - P.zza Gramsci 3 - 20145 Milano - Tel. 02/317676

Ivo Trigari - V.le Kennedy 64 - 81055 S. Maria C. V. (Le)

Enzo Coassin - Via Dante 42/C - 33085 Maniago (Pn) - Tel. O427/72691

Franco Carta - Str. 6 Luri - 09092 Arborea - Tel. 0783/489919

Remo Ghezzi - P.zza S. Francesco 10 - 52043 Castiglion Fiorentino (Ar) - Tel. 0575/658685

64 User Club - Via Roma 10 - 80049 Somma Vesuviana (Na)

Enrico Antinozzi - Corso Europa 26 - 80127 Napoli

Renato Clementi - Via Repubblica 64 - 28026 Omegna (No) - Tel. 0323/641354

Emilio Frongia - Via Brighindi 179 - 03100 Frosinone - Tel. 0775/852495

Stefano Ferreri - C.so Grosseto 259 - 10147 Torino - Tel. 011/296892

Luca Giustozzi - Via Pausola 113 - 62014 Conidonia (Mc) - Tel. 0773/292453

Carlo Pezza - Via S. di Santarosa 61 - 00149 Roma - Tel. 06/5281016

Massimiliano Tassoni - Via Mazzini 82 - 40138 Bologna - Tel. 051/344405

Piero Lacaita - Via Palestro 50 - 74020 Torricella (Ta) - Tel. 099/653049

Giuseppe Pini - Via Bezzecca 9 - 61100 Pesaro Tel. 0721/32111

Riccardo Bossi - Via G. Imperatore 15 - 00145 Roma - Tel. 06/5134603

Mauro Spreafico - Via Solari 41 - 20144 Milano - Tel. 02/4227730

Maurizio Guidato - Via Putignano 26 - 56014 Pisa - Tel. 050/982281

Francesco Liperati - Via A. Grandi 22 - 22040 Sirone - Tel. 031/850713

Alfonso Forgione - Via Cavaliere di Vittorio Veneto 28 - 83040 Gesualdo - Tel. 0825/401221

Sandro Natali - Via 14 Luglio 36/A - 50019 Sesto Fiorentino (Fi) - Tel. 055/4481640

Alessandro Sciortino - Via Galileo Galilei 22 - 90145 Palermo - Tel. 091/568039

Rosario Melilli - Via B. Vittone 22 - 10023 Chieri (To) - Tel. o11/9424950

Gian Foresi - Via Vaina 2 - 20122 Milano - Tel. 02/592459

Gianluca Brotto - Via Serra di Falco 149 - 90145 Palermo - Tel. 091/574554

Zac Soft - Via Unione Sovietica 15 - 50126 Firenze - Tel. 055/680806

Davide Baroni - Via Pezzana 9 - 40127 Bologna - Tel. 051/516529

Matteo Doveri - Via I Maggio 15 - 56025 Pontedera - Tel. 0587/212154

Tommaso Masi - Via Toscanini 17 - 35031 Abano Terme (Pd) - Tel. 049/810881

Enzo Meocci - Via Anconetana 4 - 06016 S. Giustino (Pg) - Tel. 075/856676

Pietro Coletta - Via Oriolese/loc. Montevirginio 25 - 00060 Canale Monterano - Tel. 06/9027228

Massimiliano Bartolozzi - Via Vecchia Napoli Km. 4 - 00049 Velletri - Tel. 06/9613370

Giuseppe Monticelli - Via XXV Aprile 98 - 20029 Turbigo (No) - Tel. 0331/899218

Maurizio Corvi - Via Alfieri 88 - 03024 Ceprano - Tel. 0775/950813

Roberto Baccetti - Via Rosmini 7 - 56100 Pisa Tel. 050/44627

Massimo Siena - Via Centurini 15 - 05100 Terni - Tel. 0744/59497

Roberto Quaglia - Via Martinazzoli 2 - 20161 Milano - Tel. 02/6462130

Stefano Guerriero - Via F. Grossigondi 45 - 00167 Roma - Tel. 06/8323519

Assunta Lara - Via Predosa 11 - 40069 Zola Predosa (Bo) - Tel. 051/754401

Pierpaolo Monfardini - Via G. Sozzi 15 - 40033 Casalecchio (Bo) - Tel. 051/573393

Stefano Benso - C.so A. Podestà 12 - 16128 Genova - Tel. 010/540471

Alfonso Micucci - Via B. Croce 172 - 65100 Pescara - Tel. 085/66891

Pino De Palma - Q.re S. Teodoro 22/3 - 30173 Mestre (Ve) - Tel. 041/5058376

Dino Marocchi - Via Marconi 302 - 65100 Pescara - Tel. 085/68352

Gaetano Sicurella - Via Paolo Bentivoglio 20 - 95125 Catania - Tel. 095/334402

Andrea Ciani - Via Edoardo Maragliano 32 - 00151 Roma - Tel. 06/5376977

Tommaso Gallo - Via Papa Giovanni XXIII 32 - 84013 Cava de' Tirreni - Tel. 464714

G. Riccardi/C. Mastracco - Via Carbonaro 55 03024 Ceprano (Fr) - Tel. 0775/94555

INVIARE TUTTA LA PAGINA ANCHE SE SI UTILIZZA UNA SOLA SCHEDA

Nome .... Cognome ....

Via .... N° .... CAP. .... Città ....

Telefono .... Orario ....

Registrate il mio abbonamento annuale a Commodore Computer Club.

☐ Ho versato oggi stesso il canone di L. 28.000 a mezzo c/c postale n°37952207 intestato a:
Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

☐ Ho inviato oggi stesso assegno bancario n...
per l'importo di L. 28.000 intestato a Commodore Computer Club.

Si prega di scrivere il proprio nome e l'indirizzo completo in modo chiaro e leggibile. Inviare la fotocopia del bollettino di c/c postale.

Considerando che i numeri 1, 2 e 7 sono esauriti, vogliate inviarmi i numeri arretrati .... al prezzo di L. 5.000 cadauno per richieste fino a 4 numeri. o di L. 4.000 cadauno per richieste oltre i 4 numeri arretrati, e perciò per un totale di L............... Sono a conoscenza che i fascioli suddetti non saranno inviati in contrassegno e, pertanto, ho provveduto oggi stesso a versare il canone di L.................. a mezzo c/c postale n. 37952207 intestato a:
Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

STATISTICA

Non posseggo un computer ☐

Posseggo un C64 .... si ☐ ... no ☐
Posseggo un VIC 20 .... si ☐ ... no ☐
Posseggo un Commodore Plus 14 .... si ☐ ... no ☐
Posseggo un Commodore Plus 16 .... si ☐ ... no ☐
Posseggo un registratore dedicato .... si ☐ ... no ☐
Posseggo un drive 1541 .... si ☐ ... no ☐
Posseggo una stampante .... si ☐ ... no ☐
Posseggo un monitor .... si ☐ ... no ☐

COLLABORAZIONE

A titolo di prova vi invio un articolo e la cassetta.........................disco.................................
col programma che intendo proporre per la pubblicazione di cui garantisco l'originalità.

DOMANDA/RISPOSTA ....

....

....

....

....

....

RICHIESTA ARGOMENTI

Mi farebbe piacere che Commodore Computer Club parlasse più spesso dei seguenti argomenti:

1/ ....

2/ ....

3/ ....

4/ ....

GIUDIZIO SUI PROGRAMMI DI QUESTO NUMERO

Ho assegnato un voto da 0 a 10 ai programmi che indico di seguito:

A/ .... Voto ....

B/ .... Voto ....

C/ .... Voto ....

D/ .... Voto ....

PICCOLI ANNUNCI..

....

....

....

....

....

CERCO/OFFRO CONSULENZA..

....

....

....

....

....

INVIARE IN BUSTA CHIUSA E AFFRANCANDO SECONDO LE TARIFFE VIGENTI A:

COMMODORE COMPUTER CLUB

V.le Famagosta, 75
20142 Milano

INVIARE TUTTA LA PAGINA ANCHE SE SI UTILIZZA UNA SOLA SCHEDA

Nome .... Cognome ....

Via .... N° .... CAP. .... Città ....

Telefono .... Orario ....

Non tutti i leoni sono veramente Leoni.

Ecco come riconoscere un vero programma Leoni Informatica

COMMODORE 64 SOFTWARE

Leoni informatica

Garanzia a vita
Anche dopo vent'anni un programma che rivelasse un difetto d'origine viene subito sostituito.

Assistenza telefonica
Una centralina telefonica risponde ad ogni chiamata. Leoni Informatica ti fornisce anche questo servizio assicurandoti tutte le informazioni che ti necessitano per la perfetta efficienza del tuo sistema.

Aggiornamento continuo
Il mondo si evolve e le necessità cambiano. Solo per questo anche i nostri programmi possono invecchiare. Noi te li sostituiamo aggiornati.

Quando per il tuo home-computer il negoziante ti offre un programma a basso costo, diffidane. Nella quasi totalità dei casi si tratta di una copia duplicata, che per di più può non girare bene. Le conseguenze, specialmente se si tratta della tua contabilità, sono facilmente immaginabili. Leoni Informatica, Azienda leader, fa programmi da sempre, e da sempre è sinonimo di altissima qualità. Riconoscere questi programmi è facile. La classica confezione bianca e blu è accuratamente sigillata. All'interno, allegate al floppy disk, vi sono le istruzioni in italiano e, cosa importantissima, la cartolina di garanzia. Inoltre Leoni Informatica è stata la prima in Italia ad offrire la garanzia a vita, l'assistenza ed il continuo aggiornamento dei suoi programmi. Al tuo negoziante chiedi quindi la qualità, l'assistenza e l'aggiornamento: chiedi i programmi Leoni Informatica.

Richiedi a Leoni Informatica l'elenco guida ai suoi programmi.

BES Milano

Leoni Informatica non ti abbandona mai

Leoni informatica S.r.l. - Sviluppo Software - Vendita Hardware
Via Valsolda, 21 - 20143 Milano - Tel. 02-8467378-8465072

Potete trovare i nostri programmi in tutta Italia nei punti vendita Melchioni e dai rivenditori autorizzati che espongono il nostro marchio.